Studies on Queueing Models with Vacations and Their Applications(Dissertation_全文)

AUTHOR(S):

Kasahara, Shoji

STUDIES ON QUEUEING MODELS WITH VACATIONS AND THEIR APPLICATIONS

SHOJI KASAHARA

DECEMBER 1995

STUDIES
ON
QUEUEING MODELS WITH VACATIONS AND THEIR APPLICATIONS

SHOJI KASAHARA

DECEMBER 1995

To Mayako

STUDIES
ON
QUEUEING MODELS WITH VACATIONS AND THEIR APPLICATIONS

by

SHOJI KASAHARA

Submitted in Partial Fulfillment of
the Requirement of the Degree of
DOCTOR OF ENGINEERING

Applied Systems Science

KYOTO UNIVERSITY
KYOTO 606-01, JAPAN

DECEMBER 1995

Preface

A variety of queueing models have been proposed and analyzed to evaluate the performance of systems such as computer, communication and manufacturing systems. Among them, queueing systems with vacations have been extensively studied in the last two decades since those have a lot of applications in those real systems.

For example, in most computer systems, a processor is shared among various types of jobs and hence is not available all the time to each type of jobs. From the view point of a specific job type, it alternately handles jobs of the type or jobs of the other types. To reflect the occasional inavailability of the processor in queueing systems, the server is regarded to take vacations.

Although a large number of works for the queueing systems with vacations have been carried out, there are still unsolved problems expressed by service disciplines, buffer control policies and the non-Poissonian arrival process. In this dissertation, we study such queueing models with vacations. We focus our attention to a finite parameter which characterizes the way of service and effects the system performance.

First of all, we consider queueing systems with vacations and a finite buffer under three service disciplines and study the difference among those waiting time distributions in detail. Three disciplines considered here are (1) first-come first-served (FCFS), (2) random scheduling and (3) last-come first-served (LCFS).

Secondly, we analyze the $M/G/1/K$ system with vacations under the buffer control policy called push-out scheme and investigate how the buffer policy effects the waiting time distribution. We consider following two buffering policies: Non-Preemptive-Buffering (NPB) and Preemptive-Buffering (PB), and investigate the mean waiting time and the coefficient variation of the waiting time for each policy.

Finally, we focus the queueing models with vacations in which the arrival process is not Poissonian. Most of the previous works on vacation models have assumed that customers arrive to the system according to a stationary Poisson process. The assumption of Poisson arrivals is fit to model the arrival process of data messages, and performance measures, such as the mean waiting time, are given by simple formulas.

According to the evolution of the communication technology, however, such diverse traffic as packetized voice and video can be integrated into data networks. Poisson process may not be suitable to describe bursty traffic such as voice and video, where there exits a fair amount of correlation and variation. Thus, queueing models with non-Poissonian arrivals are of much current interest in these days.

Concerning the non-Poissonian arrival process, we study the following queueing models: $SPP/G/1$ with vacations and E-limited service discipline, and $MAP/G/1$ queues under N-policy with and without vacations. A Switched Poisson Process (SPP) is a two-state Markov Modulated Poisson Process ($MMPP$) and some performance measures can be derived explicitly. On the other hand, Markovian Arrival Process (MAP) is a fairly general process and has a capability of representing a wide class of arrival processes. In both models, we investigate the effects of

the arrival process for the waiting time.

The results of this dissertation are fairly fundamental for the queueing theory. The author expects that those results will be widely useful to resolve the problems which arise in modeling computer, communication and manufacturing systems. He also hopes that this work is helpful for the further research in the performance evaluation field.

December 1995

Shoji Kasahara

Acknowledgment

I would like to express my profound gratitude to Professor Toshiharu Hasegawa of Kyoto University for his persistent encouragement and liberal supervision. He gave me a number of invaluable visions for this active and exciting field. With his enthusiastic guidance and constant support, I could accomplish this work.

I am heartily grateful to Associate Professor Yutaka Takahashi of Kyoto University for his invaluable advises and insightful suggestions on my work. He led me to the performance evaluation field of the computer communication systems in a wide scope, and taught the queueing theory as the elegant and powerful tool for this field. He also offered me lots of chances and opportunities for accomplishing my work.

I would like to express my profound appreciation to Associate Professor Tetsuya Takine of Osaka University for his stimulating discussions and valuable comments about mathematical and computer problems. It is he that taught me the queueing theory and its applications in detail and discussed the current topics of this exciting field with me.

I wish to express my special thanks to Professor Hideaki Takagi of University of Tsukuba for his helpful advises and comments. He gave me invaluable suggestions of this work and did a joint research with me.

Thanks are in order to Assistant Professor Hiroyuki Kawano of Kyoto University, Doctor Yutaka Matsumoto and all of my friends and colleagues in Professor Hasegawa's Laboratory for their encouragement.

I would like to thank my parents, Enji and Masaye Kasahara, for their understanding and encouragement of my studies.

Finally, I would like to express my sincere gratitude to Mayako by dedicating this work.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A variety of queueing models have been proposed and analyzed to evaluate the performance of systems such as computer, communication and manufacturing systems. In most of these systems, the server is busy when there are at least one customer in the system and the server becomes idle when there are no customers in the system. The server may start its service just after the customer arriving at the system.

In the case of a computer system, the processor has to do a lot of works such as a job scheduling, a process management, maintenance works for troubles, etc. From the specific job's point of view, the service to other processes can be considered as a vacation. According to the service or priority types, we can classify customers into two types, primary and secondary customers. Queueing systems in which the server works on primary and secondary customers arise naturally as models of computer, communication and production systems. As far as the primary customers are concerned, the server working on the secondary customers is equivalent to the server taking a vacation and not being available to the primary customers during this period. Thus, there is a natural interest in the study of queueing systems with server vacations.

Queueing models with vacations are also useful for analyzing the system with priorities. There are a number of works for queueing systems with priority rules. A priority rule determines the allocation of resources to customers. This rule may take into account other factors such as differences in delay costs or service times among customers classes. These factors may be explicitly modeled, or may be implicit in the assignment of customers to priority classes, where some classes receive better service than others. Main results for those priority models are devoted to first moment of performance measures such as queue length and waiting time. Unfortunately, distributions and higher moments of such performance measures are often given by complex expressions. In many cases, the vacation formulation for models with a priority are very useful for finding these other measures.

This dissertation studies single server queueing systems with vacations mainly focusing on the following subjects: finite buffer, buffer control policies and non-Poissonian arrival process.

Most of previous works have assumed that the system has an infinite buffer since this assumption enables us to derive simple and elegant formulas for performance measures such as the queue length distribution and the waiting time distribution. However, practical systems only have a finite buffer and if we need to investigate the behavior of these systems in detail, the assumption of a infinite buffer is not suitable. The analysis of systems with finite buffer is necessary for this situation. In this dissertation, we consider queueing systems with vacations and finite buffer under several service disciplines.

Secondly, we consider queueing systems with vacations under several buffer control policies.

In general, the buffer control scheme is an important factor for processing messages as well as service disciplines. Buffering policies specify those messages that are admitted to enter and those to be removed from the buffer instead when the buffer is full. It is important to study the buffer behavior under several control policies.

Finally, we study queueing systems with vacations and a non-Poissonian arrival process. From a number of works for the Broadband ISDN (B-ISDN) and its related technology, Asynchronous Transfer Mode (ATM), it has been pointed out that the Poisson arrival assumption is not suitable for modeling the traffic composed of different kinds of data packets called cells due to its burstiness property.

In addition to the ATM, we can find several applications in the manufacturing and inventory systems, where there exists the correlation in arrivals and hence the arrival stream cannot be modeled by a Poisson process. Thus, it is greatly significant to investigate how arrival processes effect performance measures such as mean waiting time.

Throughout the dissertation, service requests such as customers and jobs are called as *messages* if there are no specifications for the service requests.

The remainder of this chapter is organized as follows. In section 1.1, we show typical vacation disciplines considered in this dissertation. In section 1.2, we present some important practical examples and show how to express those applications by queueing models with vacations. In section 1.3, we summarize the non-Poissonian arrival processes dealt in the dissertation. In section 1.4, we show the previous works of queueing systems with vacations. Finally, we present the overview of this dissertation in section 1.5.

1.1 Principal Vacation Disciplines

There are a number of combinations of service and vacation disciplines. In this section, we show some principal vacation disciplines and service policies. Considering the state at the end of a busy period, we can classify vacation disciplines into two categories, *exhaustive service* and *non-exhaustive service* [Taka91]. Exhaustive service implies that a vacation begins only when there are no messages in the system. On the other hand, under non-exhaustive service, a vacation begins although there are messages in the system. First, we show some vacation disciplines under exhaustive service.

- *Multiple Vacations*
  We assume that the server begins a vacation each time the system becomes empty. If the server returns from a vacation and find the system not empty, it starts to work immediately and continues until the system becomes empty again. If the server returns from a vacation to find no messages waiting, it begins another vacation immediately, and continues in this manner until he finds at least one message waiting upon returning from a vacation.

- *Single Vacation*
  The server takes exactly one vacation immediately after the end of each busy period. If he finds no message in the system upon returning from the vacation, it becomes idle until a message arrives. When a message arrives at the system, the server immediately starts to serve it.

- *N-policy without Vacations*
  At the end of a busy period, the server is turned off and inspects the queue length every time a message arrives. When the queue length reaches a pre-specified value N, the server

turns on and serves messages continuously until the system becomes empty. (In [Taka91], this is referred to as N-policy.)

- *N-policy with Vacations*
  At the end of a busy period, the server takes a sequence of vacations. At the end of each vacation, the server inspects the queue length. If the queue length is greater than or equal to a pre-specified value N at this time, the server begins to serve messages continuously until the system becomes empty. (In [Taka91], this discipline is referred to as *vacations with a threshold.*)

Next, we show non-exhaustive service cases as following.

- *Gated Service*
  When the server returns from a vacation, it accepts and serves continuously only those messages that are waiting at that time, deferring the service of all messages that arrive during the service period until after the next vacation. There are some variations of gated service systems. For example, in the multiple vacation model, if the server returns from a vacation to find no messages waiting, it begins another vacation immediately, and continues in this manner until it finds at least one message waiting upon returning from a vacation. In the single vacation model, the server takes exactly one vacation after the end of each busy period.

- *Limited Service*
  In the limited service, the number of messages that are served continuously during a service period is limited. Similar to the gated service, there are variations of the limited service.

  - *Pure Limited Service*
    The server takes a vacation each time it completes service to a message.
  - *G-limited Service*
    Let M be a positive integer and L_n^* denote the number of messages found in the system when the server returns from the nth vacation. Then, the server continues to serve $\min[M, L_n^*]$ messages during a service period, and then takes the next vacation. Note that the case $M = 1$ corresponds to the pure limited service and that the case $M = \infty$ corresponds to the gated service.
  - E-limited Service
    The server continues to serve until (1) M messages (including new arrivals) are served, or (2) the system empties, whichever occurs first. Note that the case $M = 1$ reduces to the pure limited service and that the case $M = \infty$ corresponds to the exhaustive service.
  - B-limited Service
    Messages are served in batches of a fixed size M. The server takes a vacation following the completion of a service period for each batch. If the server finds fewer than M messages queued upon returning from a vacation, he takes another vacation, and continues to operate in this manner until he finds at least M messages queued upon returning from a vacation.
  - T-limited Service
    The length of each busy period is limited by a given time.

In this dissertation, multiple vacations and exhaustive service are considered in Chapters 3 and 4, multiple vacations and E-limited service discipline in Chapter 5, and N-policy with and without vacations in Chapter 6.

1.2 Examples

In this section, we show some vacation models presented in [Dosh86].

There are a variety of problems and questions which can be addressed by using appropriate vacation type models. These problems are from a diverse mix of application areas. Some of these examples illustrate end applications, while others show how vacation models arise in other well-known queueing models from a broad range of applications.

1.2.1 Machine Breakdowns

A machine producing a variety of items (primary jobs) can be modeled as a single-server queue. Machine breakdown may occur randomly, independent of the status of the queue, and may be regarded as secondary jobs which preempt the primary jobs. Alternatively, breakdowns may be regarded as server vacations. The natural question here is how breakdowns affect the capacity of the machine, the queue length and the sojourn time of primary jobs (items being produced). The system is also equivalent to a two-priority single-server queue with breakdowns having a preemptive priority over the primary jobs. The vacation models are closely related to the priority models.

1.2.2 Maintenance in Production Systems

When a machine becomes idle, preventive maintenance starts. While it is in process, any items arriving at the machine will have to wait. A period for maintenance can be considered as a vacation. However, the start of vacation depends on the state of the queue. It happens only when the queue becomes idle after a busy period. Moreover, there is exactly one vacation after the end of each busy period. This is a typical example of the single vacation model.

In this situation, our main interests are how preventive maintenance affects the waiting time of the primary jobs, and how long each duration for preventive maintenance should be scheduled after the end of each busy period.

1.2.3 Maintenance in Computer and Communication Systems

Processors in computer and communication systems do considerable testing and maintenance besides their primary functions (processing telephone calls, processing interactive and batch jobs, receiving and transmitting data, etc.). The testing and maintenance are mainly to preserve the normality of the system and to provide high reliability. The way these functions are scheduled relative to the primary jobs depends on the system requirements on the delays for the primary and maintenance functions. A few illustrative situations are the following:

1. Frequently, the maintenance work required is divided into short segments. Whenever the primary jobs are absent, the processor does a segment of the maintenance work. If, on completion of this segment, some primary jobs are present, then the processor will serve the primary jobs until it is idle again. On the other hand, if no primary job is present on completion of a maintenance segment, then a second maintenance segment is done and so on. Here, maintenance is the lowest priority work done in short segments. Primary jobs have non-preemptive priorities over the maintenance segments. Also, various types of maintenance segments are arranged in a cyclic sequence and when the entire sequence is completed once, the cycle repeats. Thus, while primary jobs are being served, the system behaves like a usual queueing system. When the system is idle, the server takes a vacation

(works on a maintenance segment) and keeps on taking vacations until, on return from a vacation, the server finds at least one primary job waiting. This can be considered as the multiple vacation model.

Since, in this system, an unusually heavy load of primary jobs may shut-off maintenance for a prolonged period, some measures are frequently taken to guarantee that a certain minimum amount of maintenance work will be done in a given interval. One such measure is to monitor the amount of time available to the maintenance work and limit the acceptance of primary jobs so that the required amount of time is available for maintenance.

2. A limit M is placed on the number of primary jobs done before at least one segment of maintenance work is done. The resulting queueing model is a limited service vacation model in which the server takes vacation on becoming idle or after serving M consecutive primary jobs.

3. This is similar to 2 but the limit is placed on the time T spent on primary jobs rather than the number of primary jobs served.

4. Maintenance jobs are scheduled periodically and, when scheduled, they get preemptive or non-preemptive priority over the primary jobs. The vacations are secondary jobs which arrive independently of the state of the system and have priority over primary jobs. The resulting queueing model is similar to that in 1.2.1.

Typical questions here are the effects of maintenance jobs on the delay of primary jobs, the appropriate length of a maintenance segment, and the appropriate values of the limits M and T in the limited service case.

1.2.4 Cyclic Server Queues

Cyclic server queues arise naturally as models with scheduling task processing in a variety of computer systems, and those under disciplines by which various contending ports or virtual circuits are served in communication systems. There are a number of works on cyclic server queues, dealing with both the fundamental analysis and applications to computer and communication systems. Here, we only briefly describe the basic models and discuss their relationships to the vacation models.

The basic model has m classes of messages, each with its own queue. These m queues are served by a single server. The single server serves these queues in a cyclic way according to a specified order. At time 0, the server visits the first queue on the template. After completing specified work there, it moves to the second queue in the template and so on until it completes work in the last queue in the template. At this point it goes back to the first queue and next cycle starts again. Various models of this type are distinguished by when the server decides to move from one queue in the template to the next. In an *exhaustive service* case, the server leaves a queue when it is empty. In the *gated service* case, the server, on arrival to a queue, closes a gate behind the waiting messages in that queue and leaves that queue when the messages present inside the gate are served. Finally, in the *limited service* case there is a limit M_i placed on the number of messages served on each visit to queue i. The server leaves queue i either when that queue is empty or when M_i messages have been served during the current visit.

Two different types of vacation models are related to the cyclic server queues. Considering a specific queue we note that, as far as the messages in that queue are concerned, the time the server spends serving other queues as well as moving from/to queues is like a vacation. In

exhaustive service, this vacation begins when the queue in question is idle and the vacation model is of a multiple vacations type. In the gated or limited service case, vacation may start even when messages are present in the queue, but only on completion of a service. However, unlike the models discussed earlier, the length of each vacation here depends on the number of arrivals in other queues since the last visit of the server to each of those queues. This, in particular, implies that vacation time is strongly dependent on the length of busy periods. The distribution of vacation time is not known a priori. Consequently, all the results cannot be necessarily obtained from the known results for vacation models. However, vacation models can be and have been used successfully to obtain either iterative procedures or approximations for the cyclic server queues. In some cases, the results from vacation models are also used to obtain exact expressions for various performance measures.

When the underlying system is completely symmetric, that is all the interarrival, service time and the walk time distributions are independent of the index of the queue, then the average waiting time can be obtained directly from the total number of messages in the system. Here, a different type of vacation model would be appropriate. Suppose we consider a single queue consisting of all the waiting messages. Vacations are then the time intervals corresponding to the movements of the server from one queue to another. Vacation starts even when there are messages in the queue (even in the exhaustive service case). However, the vacation distributions are explicitly known and their durations are independent of the arrival processes. This makes the application of the results from vacation models relatively easier.

1.2.5 Clock Driven Schedules

These types of schedules are frequently used in computer systems for call processing applications to schedule primary and maintenance work. We present two variants depending on how the clock is used. In both cases there is a clock which ticks every T seconds. Primary jobs arrive to join an external queue and the clock ticks are used to decide when these jobs are moved to an internal queue from which they get served. Moreover, there is an unending supply of maintenance work divided into segments as discussed earlier.

1. At each clock tick, all the primary jobs waiting in the external queue are moved to the internal queue where they have a non-preemptive priority over the maintenance work. Arrivals between the clock ticks wait in the external queue until the clock ticks. If we concentrate on the internal queue of primary jobs, then interarrival times are constant (equal to the interval between clock ticks) and when this queue becomes empty, the server takes a vacation to do a maintenance segment and keeps on doing these segments until the clock ticks. If new primary work arrives at the internal queue at this point, after the current maintenance segment, the primary busy period starts again. We thus have a $D/G/1$ queue with multiple vacations.

2. The clock is asynchronous to the basic arrival and service processes. After completing each primary job, the server checks the external queue and brings any waiting primary jobs for service. Thus, as long as the primary job queue is non-empty, this behaves like a usual queuing system. When the primary queue is empty, a maintenance segment is started and continued until the asynchronous clock ticks again. At this point the primary queue is checked again. If a primary job is present, it gets preemptive priority over the maintenance work in progress, but otherwise the maintenance work is continued until the next clock tick and so on. Thus, after the end of each busy period the server takes a vacation until the next clock tick (the length of this vacation is random with support on $[0, T]$) and

keeps on taking vacations (subsequent vacations are of constant length T) until, on return from a vacation, it finds at least one message present. Then a busy period starts again. This model differs from the previous models in that the first vacation after the end of each busy period has a different distribution than the subsequent ones. In general, we may have a sequence of (not necessarily identical) distributions which govern the vacations after the end of each busy period. We will call this a queue with variable vacations.

1.2.6 Priority Queue

Consider a queueing system with multiple vacations as described in 1.2.3 above. Now consider a queueing system with two priority classes, the high priority messages and low priority ones. In this priority queueing model, if the total load approaches 1, the low priority queue is always full and, if the priority is non-preemptive, the high priority queue behaves just like the queueing system with vacations.

For simplicity, suppose that we have a priority queueing system with three priority classes and we are interested in the messages with intermediate priority. The high priority messages play the role of interruptions or breakdowns in service. These may occur during an ongoing service of class 2 messages or at the end of such a service, depending on whether the service to higher priority messages is preemptive or non-preemptive. In the case of non-preemptive service, when the lowest priority messages go into service, those play the role of vacations which start only when the class 2 (and class 1) queue is empty.

1.2.7 Related Models

Various other situations where the server is not always available to serve its primary jobs may look different but are closely related to the vacation models. In many of these cases the results from the vacation models can be successfully applied with a little more effort. In others, essentially the same techniques can be used to analyze from scratch. Now, we discuss queues with set-up time.

These arise in many production systems where each run involves set-up during which the machine is not available for productive work. If the type of set-up required is not known before the first arrival, the set-up for the service starts when this arrival occurs and the service starts after the set-up ends. This can be formulated as a vacation model in which vacation begins when an idle period ends. Here vacation starting epochs are dependent on the arrival process. On the other hand, the vacation models discussed earlier can be formulated as set-up models where the set-up time is the remaining length of the vacation in progress when an arrival finds the system empty. In any case, vacation and set-up time models are closely related, and in turn, both are related to the priority queueing models.

A related situation is one in which the first job to start a busy period has a service time distribution different from the others. The set-up time model is, in a sense a special case of this situation where the first service is the sum of set-up time and regular service time. The difference is that here the waiting time of the first job is zero, while in the set-up model it is not.

1.3 Non-Poissonian Arrival Process

In this section, we present the non-Poissonian arrival processes dealt in the dissertation.

Non-Poissonian arrival processes have been studied in the context of the performance evaluation of B-ISDN, especially ATM. ATM is based on packet oriented information transfer using small, fixed size blocks called cell and statistical multiplexing [Turn86, Armb87, Part94]. It makes possible efficient transmission of bursty traffic, such as packetized voice, image and video generated from various terminals.

Since the traffic stream in ATM network has the property of burstiness, it is hardly enough to model such correlated traffic using a Poisson process. Thus, non-Poissonian arrival processes become more important to model the system with bursty traffic. In the following, we present some non-Poissonian arrival processes in detail.

1.3.1 Markovian Arrival Process

Markovian Arrival Process (*MAP*) is one of the most useful stochastic process among non-Poissonian arrival processes. In the following, we summarize the *MAP* represented by (C, D) [Luca90].

We consider a Markov process on the state space $\{1, 2, \cdots, m+1\}$, where $\{1, 2, \cdots, m\}$ are transient states and $\{m+1\}$ is absorbing. Assume the Markov process is in a transient state i, $1 \leq i \leq m$. The sojourn time in this state is exponentially distributed with parameter λ_i. When the sojourn time has elapsed, there are two possibilities. With probability p_{ij}, $1 \leq j \leq m$, the Markov process enters the absorbing state and is instantaneously restarted in the transient state j. With probability q_{ij}, $1 \leq j \leq m$, $j \neq i$, the process immediately enters the transient state j. We define C_{ij} and D_{ij} as

$$C_{ij} = \lambda_i q_{ij},\ 1 \leq i, j \leq m,\ i \neq j, \qquad C_{ii} = -\lambda_i, \qquad D_{ij} = \lambda_i p_{ij},\ 1 \leq i, j \leq m.$$

Let C (D) denote the matrix with elements C_{ij} (D_{ij}). We note that the assumption that absorption is certain, starting from any transient state, is equivalent to the non-singularity of matrix C. The *MAP* with (C, D) is a semi-Markovian arrival process and the probability density function (pdf) for the lengths of interarrival times is given in a matrix form:

$$f(x) = e^{Cx} D. \tag{1.1}$$

The irreducible matrix $C + D$ is the infinitesimal generator of the Markov process restricted to the states $\{1, \cdots, m\}$. Let π denote the stationary vector of $C + D$, i.e.

$$\pi(C + D) = \mathbf{0}, \qquad \pi \mathbf{e} = 1, \tag{1.2}$$

where $\mathbf{e}$ denotes the column vector of ones.

Let $N(t)$ be the number of arrivals in $(0, t]$ and $J(t)$ the state of the Markov process at time t. Define the following conditional probabilities:

$$P_{ij}(n, t) = Prob\{N(t) = n,\ J(t) = j \mid N(0) = 0,\ J(0) = i\}, \qquad 1 \leq i, j \leq m.$$

We define $P(n, t)$ as the $m \times m$ matrix with elements $P_{ij}(n, t)$. $P(n, t)$ satisfies the following forward Chapman-Kolmogorov equations:

$$\begin{aligned} \frac{d}{dt} P(n, t) &= P(n, t)\, C + P(n-1, t)\, D, \qquad n \geq 1,\ t \geq 0, \\ P(0, 0) &= I, \end{aligned}$$

where I represents the unit matrix. We define the matrix generating function $P^*(z, t)$ as

$$P^*(z, t) = \sum_{n=0}^{\infty} P(n, t)\, z^n.$$

Then, $P^*(z, t)$ is given by

$$P^*(z, t) = e^{(C+zD)t}, \qquad |z| \leq 1, t \geq 0. \tag{1.3}$$

The fundamental arrival rate of this process is given by $\lambda = \pi De$.

1.3.2 Markov Modulated Poisson Process

Markov Modulated Poisson Process (*MMPP*) is a doubly stochastic Poisson process and can be constructed by varying the arrival rate of a Poisson process according to an m-state irreducible continuous time Markov chain which is independent of the arrival process [Fisc92]. When the Markov chain is in state i, arrivals occur according to a Poisson process of rate λ_i. The *MMPP* is parameterized by the m-state continuous-time Markov chain with infinitesimal generator Q and m Poisson arrival rates $\lambda_1, \cdots, \lambda_m$. Let $\Lambda = diag(\lambda_1, \cdots, \lambda_m)$. Using the *MAP* notations, we have

$$C = Q - \Lambda, \qquad D = \Lambda.$$

1.3.3 Switched Poisson Process

A Switched Poisson Process (*SPP*) is a two-state *MMPP* and hence performance measures like the queue length distribution and the mean waiting time can be derived explicitly [Taki93a, Kasa93b].

Now we consider the *SPP* which is modulated by a continuous time Markov chain with two states, 1 and 2. We assume that the time spent in state 1 (2) is exponentially distributed with rate α (β). When the state of the underlying Markov process is i, messages arrive to the system according to a Poison process with parameter λ_i.

Using the *MAP* representation, the *SPP* is expressed as follows.

$$C = \begin{pmatrix} -\alpha & \alpha \\ \beta & -\beta \end{pmatrix}, \qquad D = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

The mean arrival rate λ is given by

$$\lambda = \frac{\beta\lambda_1 + \alpha\lambda_2}{\alpha + \beta}.$$

1.3.4 Other Special Cases

In this subsection, we show some different expressions for other arrival processes using the *MAP* representation [Luca90].

- Poisson process

  Poisson process is a special case where $m = 1$ and hence the corresponding expression is:

$$C = -\lambda, \qquad D = \lambda.$$

- *PH*-renewal process

  The phase-type (*PH*) renewal process contains many familiar arrival processes including Erlang and hyperexpontial arrival processes. A *PH* renewal process with the representation ($\boldsymbol{\alpha}$, T) is expressed with *MAP* notations as

$$C = T, \qquad D = -Te\boldsymbol{\alpha}.$$

- Superposition of *MAP*s

  The superposition of two independent *MAP*s with representations (C_1, D_1) and (C_2, D_2), respectively, is also an *MAP* with

$$C = C_1 \oplus C_2, \qquad D = D_1 \oplus D_2,$$

  where $\oplus$ denotes the matrix Kronecker sum. This same construction may be extended to the superposition of $n(> 2)$ *MAP*s. The class of *MAP*s is closed under superposition.

1.4 Previous Works

In this section, we present previous works related to the dissertation. Since there are a number of researches on queueing systems with vacations and its associated models, we classify those into three parts: queueing system with vacations, buffer control policies and non-Poissonian arrival process.

1.4.1 Queueing Systems with Vacations

Since queueing systems with vacations have been the classical subject in the queueing theory, there are a number of books treating those in detail [Coop81, Taka91, Taka93, Wolf89]. In particular, [Taka91] focuses on queueing systems with vacations and infinite buffer, while [Taka93] focuses on those with finite buffer. Excellent survey papers of queueing systems with vacations, including some applications, are written by Doshi [Dosh86, Dosh90].

As for the queueing systems with vacations and finite buffer, Lee [Lee84] analyzed the waiting time of an $M/G/1/K$ system with server vacations under the exhaustive service discipline. He studied the queue length process considering the embedded Markov chain. Using a combination of the supplementary variable and sample biasing techniques, he derived the general queue length distribution of the time continuous process, the blocking probability, and the waiting time distribution. Lee [Lee89a] also studied an $M/G/1/K$ with vacations and limited service discipline in the similar manner to [Lee84].

1.4.2 Buffer Control Policies

Buffer control policies specify those messages that are admitted to enter and those to be removed from the buffer instead when the buffer is full.

A communication system under a preemptive buffering has been investigated by Rubin and Ouaily in the context of an $M/G/1/K$ with push-out scheme [Rubi88]. They have classified buffer control policies into the following types.

- Non-Preemptive-Buffering (NPB)
  An arriving message that finds the system full is blocked.

- Preemptive-Buffering (PB)
  If an arriving message finds the system full, the message which has waited the longest is pushed out from the buffer and the arriving message is allocated a buffer space.

They considered the models of FCFS/NPB, FCFS/PB and LCFS/PB, respectively. In all cases, they derived the queue length and the waiting time distributions.

Sumita and Ozawa has studied a push-out scheme in [Sumi88] and analyzed loss probabilities and the waiting time considering the $M/D/1$ queue with finite buffer.

Takagi [Taka85] analyzed an $M/G/1/K$ where the arrival process is switched off when the buffer limit is reached, and switched on again when the buffer occupation falls below a given resume level. He derived the queue length distribution and shows numerical results of loss probability and response time.

Kröner [Krön90] has proposed a partial buffer sharing scheme which is a modified PB scheme for systems with two priorities. Class-1 messages are supposed to have the higher priority than class-2 messages. Let K_i $(i = 1, 2)$ denote the pre-specified maximum number of priority i messages in the system, and $K = K_1$. If an arriving class-2 message finds K messages or K_2 class-2 messages in the system, this class-2 message cannot enter the system. When a class-1 message arrives at the system, the following situations are considered:

- If the system is full with class-1 messages, the arriving class-1 message cannot enter the system.

- If the system is full and there are k $(\leq K_2)$ class-2 messages, then the class-2 message which has waited the longest is pushed out from the buffer and the arriving class-1 message is allocated a buffer space.

He mainly analyzed loss probabilities and compares the numerical results with a push-out scheme.

1.4.3 Non-Poissonian Arrival Process

In this subsection, we briefly summarize the previous works for the queueing models with vacations and non-Poissonian arrival process.

In early studies of queues with non-Poissonian arrival processes, a *PH* renewal process has been mainly analyzed by Neuts [Neut79]. A *PH* renewal process is a renewal process in which inter-renewal times have a *PH* distribution. Although the notation used in [Neut79] is fairly complex, the matrix formation shows that the process is indeed a natural generalization of the ordinary Poisson process.

Ramaswami [Rama80] has introduced a N-process, which is formed from a *PH* renewal process, and analyzed the $N/G/1$ queue in detail for the first time. In [Rama80], the stationary probability distributions such as the queue length and the virtual waiting time are derived and the algorithms for calculating moments are shown in the context of the matrix analytic methodology. Neuts also developed this matrix analytic methodology in [Neut81, Neut89]. He has distinguished the matrix analytic methodology between two different paradigms: $GI/M/1$-type [Neut81] and $M/G/1$-type [Neut89], respectively.

MAP has been introduced by Lucantoni et al.[Luca90] as a generalization of *PH* renewal processes and the *MMPP*'s. In [Luca90], the representation (C, D) is introduced for the first time and a $MAP/G/1$ queue with multiple vacations are analyzed in the context of $M/G/1$ paradigm.

Blondia [Blon91] has considered a single server queue with finite buffer where the server takes vacations and analyzed the model for both the cases under the exhaustive and the limited service disciplines.

Recently, Lucantoni extended the *MAP* to a batch Markovian arrival process (*BMAP*) [Luca91, Luca93] and compared the derived formulas with Poisson cases.

In the matrix analytic approach, there are some difficulties in implementing algorithms for calculating the moments of performance measures. Takine et al.[Taki93a] considered a 2-state *MMPP* called *SPP*, and analyzed a batch $SPP/G/1$ queue with multiple vacations and exhaustive service discipline using the supplementary variable technique.

1.5 Overview of the Dissertation

Although there have been a large number of works for the queueing systems with vacations, there are still many unsolved problems in this field. We study queueing systems with vacations mainly concerning the following points: *service disciplines*, *buffer control policies* and the *non-Poissonian arrival process*. These elements characterize the way of service and hence plays an important role in the system performance.

First, we consider a queueing system with finite buffer. In this model, our main interest is the difference of waiting times under three service disciplines: FCFS, random scheduling and LCFS.

In Chapter 2, we consider an $M/G/1/K$ system without vacations under random scheduling and LCFS. We apply the results of this chapter to an $M/G/1/K$ with multiple vacations in chapter 3. We analyze the waiting time distribution under random scheduling and LCFS and compare the numerical results of the mean and the coefficient of variation of the waiting time under FCFS, random scheduling and LCFS.

In Chapter 3, we consider an $M/G/1/K$ system with vacations under random scheduling and LCFS. Applying the results obtained in Chapter 2, we analyze the waiting time distribution under random scheduling and LCFS. In numerical examples, we show the mean and the coefficient of variation of the waiting time under three service disciplines. Those numerical results are also compared with those obtained in Chapter 2.

Secondly, we consider the buffer control policies which specify the message behavior in finite buffer. In Chapter 4, we apply the results of Chapter 3 to the system with buffer control policies. We consider an $M/G/1/K$ system with push-out scheme and multiple vacations, and analyze the waiting time distribution for the message which is eventually served. Some numerical results including the comparisons between the push-out and the ordinary blocking models are presented.

Chapters 5 and 6 deal with the queueing systems with vacations under a non-Poissonian arrival process. In Chapter 5, we consider an $SPP/G/1$ queue with multiple vacations and E-limited discipline. We consider the joint probability density functions of the queue length and the elapsed service time or the elapsed vacation time. Then, we derive the equations for these probability distribution functions (PDFs) which include a finite number of unknown values. Using Rouché's theorem, we determine the values from boundary conditions and derive the transform of the stationary queue length distribution explicitly.

In Chapter 6, we consider $MAP/G/1$ queues under N-policy with and without vacations. A pre-specified value N is a finite parameter at which the server starts service after an idle period or vacations. We analyze the stationary queue length and the actual waiting time distributions

in both systems with and without vacations, and derive the recursive formulas to compute the moments of these distributions. Furthermore, we provide a numerical algorithm to obtain the mass function of the stationary queue length.

Finally, concluding remarks are provided in Chapter 7.

Chapter 2 is mainly drawn from [Kasa89], Chapter 3 from [Kasa95a], Chapter 4 from [Kasa93a], Chapter 5 from [Kasa93b] and Chapter 6 from [Kasa95b].

Chapter 2

M/G/1/K under Random Scheduling and LCFS

2.1 Introduction

This chapter considers an $M/G/1$ queue with a finite buffer. $M/G/1$ queueing systems are classical subjects and many variants of those have been studied to evaluate the performance of the computer and communication systems. In particular, Takács [Taká63] analyzed the waiting time of an $M/G/1/\infty$ system under three service disciplines: first-come first-served (FCFS), random scheduling and last-come first-served (LCFS). These three service disciplines are explained in more detail as

1. FCFS
   Messages are served in their arriving order.

2. Random scheduling
   Messages are independently selected for service regardless of their arriving order and elapsed time in the system. Messages have the uniform probability of being chosen for next service.

3. LCFS
   The message which has the least elapsed time is chosen for next service.

In this chapter, we analyze the waiting time of an $M/G/1/K$ system under random scheduling and LCFS. The subject in this chapter is to compare the performance measures under above three service disciplines.

We explain the model of an $M/G/1/K$ system in section 2.2. In section 2.3, we show the Lee's results [Lee84] of the queue length, and the joint distribution of the number of messages and of the remaining service time at an arbitrary instant. In section 2.4, we consider the length of a busy period and in section 2.5, we derive the Laplace-Stieltjes transforms (LST's) of distribution functions of the message waiting time under the two service disciplines. We show some numerical results in section 2.6.

2.2 Model

We consider an $M/G/1$ system with a finite buffer. Messages arrive at the system according to a Poisson process with a parameter λ. The PDF and the mean of the service time for a

message are denoted by $S(x)$ and b, respectively. The maximum number of messages that can be present in the system is $K < \infty$. When K messages are in the system, new arriving messages cannot enter the system. Let P_B denote the loss probability. Then, arriving messages can be accommodated in the system with probability $1 - P_B$. Throughout the chapter, we assume that the system is in the equilibrium.

2.3 Queue Length Distribution

In this section, we consider the number of messages in the system by the method of imbedded Markov chains [Coop81, Klei75, Taka89].

We choose a set of imbedded Markov points at those points in time when a service is completed. Let L_n be the number of messages in the system immediately after the nth Markov point. We define the limiting probability distribution and the state transition probabilities as

$$\begin{aligned} \pi_k &\equiv \lim_{n\to\infty} Prob[L_n = k], \qquad k = 0, 1, 2, \cdots, K-1, \\ p_{jk} &\equiv Prob[L_{n+1} = k | L_n = j], \qquad 0 \le j, k \le K-1. \end{aligned}$$

Note that L_n cannot be K because when a message leaves the system, it cannot leave behind a completely full system. At least one waiting position must be empty.

Let a_k denote the probability that there are k arriving messages in a service time. Then, we have

$$a_k = \int_0^\infty \frac{(\lambda x)^k}{k!} e^{-\lambda x} dS(x), \qquad k \ge 0.$$

We obtain state transition probabilities as

$$p_{0k} = \begin{cases} a_k, & 0 \le k \le K-2, \\ 1 - \sum_{m=0}^{K-2} a_m, & k = K-1, \end{cases} \tag{2.1}$$

and for $1 \le j \le K-1$,

$$p_{jk} = \begin{cases} a_{k-j+1}, & j-1 \le k \le K-2, \\ 1 - \sum_{m=0}^{K-j-1} a_m, & k = K-1. \end{cases} \tag{2.2}$$

The steady-state equations for state transitions are given by

$$\pi_k = \sum_{j=0}^{k+1} \pi_j p_{jk}, \qquad 0 \le k \le K-1, \tag{2.3}$$

$$\sum_{k=0}^{K-1} \pi_k = 1. \tag{2.4}$$

Substituting (2.1) and (2.2) into (2.3), we obtain

$$\pi_k = \pi_0 a_k + \sum_{j=1}^{k+1} \pi_j a_{k-j+1}, \qquad 0 \le k \le K-2, \tag{2.5}$$

$$\pi_{K-1} = \pi_0 \left(1 - \sum_{m=0}^{K-2} a_m\right) + \sum_{j=1}^{K-1} \pi_j \left(1 - \sum_{m=0}^{K-j-1} a_m\right). \tag{2.6}$$

Since (2.4) and (2.5) provide K independent equations for $\{\pi_k\ ; 0 \leq k \leq K-1\}$, we can calculate π_k's by solving these equations.

Let Π_k $(k = 0, 1, \cdots, K)$ be the probability that an arriving message finds k messages in the system. If we only consider the situation where the system is not fully occupied, the probability distribution $\{\Pi_k\}$ for the number of messages in the system immediately before arrivals is identical to the probability distribution $\{\pi_k\}$ of the number of messages in the system immediately after departures because the system state changes by unit steps only. Therefore, both $\{\Pi_k\ ; 0 \leq k \leq K-1\}$ and $\{\pi_k\ ; 0 \leq k \leq K-1\}$ satisfy the same set of equations (2.5) and (2.6), and Π_k are proportional to π_k. Thus, we have

$$\Pi_k = c\pi_k, \qquad 0 \leq k \leq K-1, \tag{2.7}$$

where c is a proportional constant. We also have the normalization condition:

$$\sum_{k=0}^{K} \Pi_k = 1. \tag{2.8}$$

In order to determine c, note that the probability distribution $\{\Pi_k\}$ of the number of messages in the system at arrival instants is identical to the probability distribution $\{P_k\}$ of that at arbitrary instant. This property comes from the assumption of a Poisson arrival process, for which we have a theorem PASTA [Wolf82].

Let γ denote the throughput of the system, ρ the offered load and ρ' the carried load respectively. Then, we have

$$\gamma = \lambda(1 - P_B), \tag{2.9}$$

$$\rho = \lambda b, \tag{2.10}$$

$$\rho' = \rho(1 - P_B). \tag{2.11}$$

Note that $P_B = \Pi_K$. From PASTA, the probability that there is no message in the system at an arriving epoch becomes

$$\Pi_0 = P_0 = 1 - \rho'. \tag{2.12}$$

From (2.11) and (2.12), we obtain

$$\Pi_K = 1 - \frac{1 - \Pi_0}{\rho}. \tag{2.13}$$

Substituting (2.7) and (2.13) into (2.8) yields

$$c = \frac{1}{\pi_0 + \rho}. \tag{2.14}$$

Using (2.7), (2.9), (2.11) and (2.14), we obtain following expressions:

$$\Pi_k = P_k = \frac{\pi_k}{\pi_0 + \rho}, \qquad 0 \leq k \leq K-1, \tag{2.15}$$

$$\Pi_K = P_K = 1 - \frac{1}{\pi_0 + \rho}, \tag{2.16}$$

$$\rho' = \frac{\rho}{\pi_0 + \rho}, \tag{2.17}$$

$$\gamma = \frac{\lambda}{\pi_0 + \rho}. \tag{2.18}$$

Next, we consider the joint distribution of the number of messages in the system, L, and the remaining service time $\widetilde{S}$ [Lee84, Taka89]. We define

$$\Pi_k^*(s) \equiv \int_0^\infty e^{-sx} Prob[L = k, x < \widetilde{S} < x + dx], \qquad 1 \le k \le K.$$

Note that

$$P_k = \Pi_k = \Pi_k^*(0), \qquad 1 \le k \le K. \tag{2.19}$$

Let $\alpha(\widehat{S})$ denote the number of messages that arrive at the system during the attained service time $\widehat{S}$. Given that the server is busy, there are k messages in the system and $\widetilde{S}$ remaining service time (1) if there are no messages at the last service completion epoch and there are $k-1$ arrivals during the elapsed service time $\widehat{S}$ of a message that arrives during the idle period, or (2) if there are $j(\le 1)$ messages at the last service completion epoch and there are $k-j$ messages during the elapsed service time $\widehat{S}$ of the next messages. Thus, we obtain

$$\begin{aligned}
\Pi_k^*(s) &= \rho'\pi_0 E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = k-1] \cdot Prob[\alpha(\widehat{S}) = k-1] \\
&\quad + \rho' \sum_{j=1}^{k} \pi_j E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = k-j] \cdot Prob[\alpha(\widehat{S}) = k-j], \qquad (2.20) \\
&\qquad\qquad 1 \le k \le K-1, \\
\Pi_K^*(s) &= \rho'\pi_0 \sum_{m=K-1}^{\infty} E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = m] \cdot Prob[\alpha(\widehat{S}) = m] \\
&\quad + \rho' \sum_{j=1}^{K-1} \pi_j \sum_{m=K-j}^{\infty} E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = m] \cdot Prob[\alpha(\widehat{S}) = m]. \qquad (2.21)
\end{aligned}$$

We define $\alpha_n(s)$ as

$$\alpha_n(s) \equiv E[e^{-s\widetilde{S}} \cdot \frac{(\lambda\widehat{S})^n}{n!} e^{-\lambda\widehat{S}}].$$

Then, we have

$$Prob[\alpha(\widehat{S}) = n] = \alpha_n(0), \qquad E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = n] = \frac{\alpha_n(s)}{\alpha_n(0)}.$$

Thus, (2.20) and (2.21) become

$$\Pi_k^*(s) = \rho' \left[\pi_0 \alpha_{k-1}(s) + \sum_{j=1}^{k} \pi_j \alpha_{k-j}(s) \right], \qquad 1 \le k \le K-1 \tag{2.22}$$

$$\Pi_K^*(s) = \rho' \left[\pi_0 \sum_{m=K-1}^{\infty} \alpha_m(s) + \sum_{j=1}^{K-1} \pi_j \sum_{m=K-j}^{\infty} \alpha_m(s) \right]. \tag{2.23}$$

$\alpha_n(s)$ is given by [Taka89] (see Appendix B in detail)

$$\alpha_n(s) = \frac{1}{\rho} \left[S^*(s) \left(\frac{\lambda}{\lambda - s} \right)^{n+1} - \sum_{m=0}^{n} a_m \left(\frac{\lambda}{\lambda - s} \right)^{n-m+1} \right], \qquad n = 0, 1, 2, \cdots. \tag{2.24}$$

Substituting (2.24) into (2.22) and (2.23), we obtain

$$\Pi_k^*(s) = \frac{1}{\pi_0+\rho}\left[S^*(s)\left\{\pi_0\left(\frac{\lambda}{\lambda-s}\right)^k + \sum_{j=1}^{k}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{k-j+1}\right\} - \sum_{j=0}^{k-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{k-j}\right], \qquad 1 \le k \le K-1, \tag{2.25}$$

$$\Pi_K^*(s) = -\frac{1}{(\pi_0+\rho)s}\left[S^*(s)\left\{\pi_0\left(\frac{\lambda}{\lambda-s}\right)^{K-1} + \sum_{j=1}^{K-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{K-1}\right\} - \sum_{j=0}^{K-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{K-j-1}\right]. \tag{2.26}$$

2.4 Busy Period

In this section, we consider the busy period [Coop81, Taka89]. Let $\bar{\theta}_K$ be the mean length of a busy period for the $M/G/1/K$ system. The state of the system regenerates with the alternating cycle of a busy period of mean length $\bar{\theta}_K$ and an idle period of mean length $1/\lambda$. Therefore, the fraction of the time that the server is busy is given by

$$\rho' = \frac{\bar{\theta}_K}{\bar{\theta}_K + 1/\lambda}. \tag{2.27}$$

Thus, the probability that an arriving message is lost is given by

$$P_B = 1 - \frac{\rho'}{\rho} = 1 - \frac{\bar{\theta}_K}{\rho(\bar{\theta}_K + 1/\lambda)}. \tag{2.28}$$

Suppose that k messages arrive during the service of the first message in a busy period of the $M/G/1/K$ system . Since the duration of a busy period is independent of the service discipline, let us assume LCFS. If $k \le K-2$, there are $K-k$ empty waiting positions at the service completion epoch of the first message. Therefore, it takes $\bar{\theta}_{K-k+1}$ in average to clear the position of the last arriving message. Similarly, it takes $\bar{\theta}_{K-k+2}$ in average to clear the position of the second to the last message, and so on. Finally, it takes $\bar{\theta}_K$ in average to clear the head of the queue. Similar arguments apply for the case of $k \ge K-1$, when the system has just one empty position at the start of the next service. So it takes the sum of $\bar{\theta}_2$, $\bar{\theta}_3$, $\cdots$ and $\bar{\theta}_K$ in average to clear all messages. Thus, we obtain the following recursive equations:

$$\bar{\theta}_1 = b, \tag{2.29}$$

$$\bar{\theta}_K = b + \sum_{k=1}^{K-2} a_k \cdot \sum_{j=K-k+1}^{K}\bar{\theta}_j + \left(\sum_{k=K-1}^{\infty} a_k\right)\left(\sum_{j=2}^{K}\bar{\theta}_j\right), \qquad K \ge 2. \tag{2.30}$$

From the above equations, we have

$$\bar{\theta}_2 = \frac{b}{a_0}, \tag{2.31}$$

$$\bar{\theta}_K = b + \sum_{j=2}^{K}\bar{\theta}_j\left(1 - \sum_{k=0}^{K-j} a_k\right), \qquad K \ge 3. \tag{2.32}$$

Following the same consideration, we find a recursive equation for $\theta_K^*(s)$, the LST of the PDF for the length of a busy period in the $M/G/1/K$ system:

$$\theta_K^*(s) = S_0^*(s)\left[1 - \sum_{k=1}^{K-2} S_k^*(s) \prod_{j=K-k+1}^{K-1} \theta_j^*(s) - \left(\sum_{k=K-1}^{\infty} S_k^*(s)\right)\left(\prod_{j=2}^{K-1} \theta_j^*(s)\right)\right]^{-1}, \tag{2.33}$$

where

$$S_k^*(s) \equiv \int_0^\infty \frac{(\lambda x)^k}{k!} e^{-(s+\lambda)x} dS(x), \qquad k = 0, 1, 2, \cdots. \tag{2.34}$$

2.5 Analysis of Message Waiting Time

In this section, we analyze the LST $W^*(s)$ of the waiting time under (1) random scheduling and (2) LCFS. We also present the previous analysis of the mean waiting time under FCFS in Appendix C.1 [Taka89].

2.5.1 Random Scheduling

The message waiting time consists of the remaining time to the next imbedded point after arrival and the duration from the imbedded point to the start of its service. To find the LST $W^*(s)$, we define $W_j(x)$ as the probability that the service of an arbitrary message among the j messages in the system starts within time x from an imbedded point.

Each message is chosen for service without delay among waiting messages (say j) with equal probability $1/j$. With probability $1-1/j$ the message is delayed for service at least one message service period. If k more messages arrive during this period, the waiting time of the message is the sum of the service period and the time whose distribution is given by $W_{j+k-1}(x)$. Therefore, we obtain $W_j(x)$ and its LST $W_j^*(s)$ as

$$\begin{aligned} W_j(x) &= \frac{1}{j} + \left(1 - \frac{1}{j}\right)\left[\sum_{k=0}^{K-j-1}\left\{\int_0^x e^{-\lambda u}\frac{(\lambda u)^k}{k!}dS(u)\right\} * W_{j+k-1}(x)\right. \\ &\qquad \left. + \sum_{k=K-j}^{\infty}\left\{\int_0^x e^{-\lambda u}\frac{(\lambda u)^k}{k!}dS(u)\right\} * W_{K-1}(x)\right], \qquad 1 \le j \le K-1, \end{aligned} \tag{2.35}$$

$$W_K(x) = \frac{1}{K} + \left(1 - \frac{1}{K}\right) S(x) * W_{K-1}(x), \tag{2.36}$$

$$W_j^*(s) = \frac{1}{j} + \left(1 - \frac{1}{j}\right)\left\{\sum_{k=0}^{K-j-1} S_k^*(s) \cdot W_{j+k-1}^*(s) + \sum_{k=K-j}^{\infty} S_k^*(s) \cdot W_{K-1}^*(s)\right\}, \qquad 1 \le j \le K-1, \tag{2.37}$$

$$W_K^*(s) = \frac{1}{K} + \left(1 - \frac{1}{K}\right) S^*(s) \cdot W_{K-1}^*(s), \tag{2.38}$$

where $*$ in (2.35) and (2.36) denotes the convolution.

If a message finds j messages in the system upon arrival and if k messages newly arrive during the remaining service time, then the waiting time of the message is the sum of the remaining

service time and the time whose distribution is given by $W_{j+k}(x)$. For simplicity, let A be the number of the messages that arrive during the remaining service time, and we define

$$\Pi^*_{j:k}(s) \equiv \int_0^\infty e^{-sx} Prob[L = j, A = k, x < \tilde{S} < x + dx].$$

Then $\Pi^*_{j:k}(s)$ becomes

$$\Pi^*_{j:k}(s) = \int_0^\infty \frac{(\lambda x)^k}{k!} e^{-(s+\lambda)x} d\Pi_j(x),$$

where $\Pi_j(x)$ is the inverse transform of $\Pi^*_j(s)$. Note that

$$\sum_{k=0}^{\infty} \Pi^*_{j:k}(s) = \Pi^*_j(s).$$

Therefore, we obtain the LST of the message waiting time for random service as

$$W^*(s) = \frac{1}{1-P_B}\left[P_0 + \sum_{j=1}^{K-1}\left\{\sum_{k=0}^{K-j-2} \Pi^*_{j:k}(s) W_{j+k}(s) + \left(\Pi^*_j(s) - \sum_{k=0}^{K-j-2} \Pi^*_{j:k}(s)\right) \cdot W_{K-1}(s)\right\}\right]. \quad (2.39)$$

2.5.2 LCFS

The waiting time of the tagged message is the remaining service time plus the length of a delayed busy period which starts with those messages that arrive during the remaining service time. Suppose that the arriving message finds j messages in the system and that there are k new arrivals during the remaining service time. The number of messages left behind in the system at the next imbedded point is $j + k$. The mean length of the busy period initiated by the last arriving message which starts with its service and ends at the beginning of the service of the last but one is $\bar{\theta}_{K-j-k+1}$. Similarly, the mean length of the busy period of the last but two is $\bar{\theta}_{K-j-k+2}$, and so on. After these periods, the service of the tagged message starts.

When the tagged message arrives at the system while the server is busy, one of the following cases arises.

1. The tagged message finds $j(\leq K-2)$ messages, and during the remaining service time
   (a) $k(0 < k < K-j-1)$ new messages arrive.
   (b) More than $K-j-1$ new messages arrive.
   (c) No message arrives.
2. The tagged message finds $K-1$ messages and new messages arriving during the remaining service time are lost.

Thus, we obtain the LST of the message waiting time as

$$W^*(s) = \frac{1}{1-P_B}\left[P_0 + \sum_{j=1}^{K-1}\left\{\sum_{k=0}^{K-j-2} \Pi^*_{j:k}(s) \cdot \prod_{l=0}^{k-1} \theta^*_{K-j-l}(s) + \left(\Pi^*_j(s) - \sum_{k=0}^{K-j-2} \Pi^*_{j:k}(s)\right) \prod_{l=2}^{K-j} \theta^*_l(s)\right\}\right]. \quad (2.40)$$

2.6 Numerical Results

In this section, We show some numerical examples.

Let μ denote the service rate of the server. Note that $b = 1/\mu$. In our numerical examples, the LST of the service time $S^*(s)$ is chosen as follows.

1. k phase Erlangian distribution:

$$S^*(s) = \left(\frac{k\mu}{s + k\mu}\right)^k,$$

where $\mu = 1.0$, and $k = 1$(exponential distribution) and 3.

2. Hyper exponential distribution:

$$S^*(s) = \sum_{i=1}^{m} \frac{k_i \mu_i}{s + \mu_i},$$

where $m = 2$, $\mu_1 = 0.5$, $\mu_2 = 3$, $k_1 = 0.4$ and $k_2 = 0.6$. In this case, the mean service time is equal to 1.0.

Let the first and second moments of the waiting time be $\bar{W}$ and $W^{(2)}$, respectively. We have calculated the following values;

1. $\bar{W}$: Mean waiting time.

2. C_W : Coefficient of variation (c.v.) of the waiting time,

$$C_W = \frac{\sqrt{W^{(2)} - (\bar{W})^2}}{\bar{W}}.$$

3. C_T : c.v. of the sojourn time in the system.

The LST of the sojourn time in the system are expressed as

$$T^*(s) = W^*(s)S^*(s).$$

Let $b^{(2)}$ denote the second moment of the service time. Then, C_T becomes

$$C_T = \frac{\sqrt{W^{(2)} - (\bar{W})^2 + b^{(2)} - b^2}}{\bar{W} + b}.$$

We have illustrated the numerical results in Figs.2.1 to 2.12.

2.6.1 Mean Waiting Time

Figs.2.1 to 2.3 show the variation of the mean waiting time for different system sizes. We can observe that the mean waiting time increases suddenly around $\lambda = 1$ and that it approaches a constant value.

The mean waiting time is independent of service disciplines, so we consider the case of FCFS. The number of messages in the system increases according to λ. However, the system size is of a finite value K, and the message that can enter the system sees at most $K - 1$ other messages. Thus, using the mean service time b $(= 1/\mu)$, the waiting time of a message is at most $(K - 1)b$. In this example, b =1.0, then each value for K =5, 10, and 20 approaches 4, 9, and 19, respectively.

2.6.2 C.V. of Waiting Time

From Figs.2.4 to 2.6, we compare c.v.'s of the waiting time under three service disciplines changing the service time distribution. In each service time distribution, the c.v. of FCFS takes the smallest value and that of LCFS takes the largest one. Under FCFS, if the tagged message arrives at the system and finds k messages ahead, its service starts certainly after the service completion of k messages. Under random scheduling, it is not sure when the service of the tagged message begin and hence the value of the c.v. is larger than that under FCFS. In the LCFS case, it is observed that the value of the c.v. diverges to infinity when the arrival rate becomes large. At the large value of the arrival rate, the waiting time of the tagged message becomes larger than that of the message which arrives after the tagged one. That is, the tagged message has few chances to be served since there are a lot of arriving messages after the tagged one. Hence, the variation of the waiting time becomes very large.

From Figs.2.7 to 2.9, we compare c.v.'s of the waiting time changing the system size K. From Figs.2.7 and 2.8, c.v.'s become small as K increases under FCFS and random scheduling. We can observe that the c.v. of $K = 20$ is the largest and that of $K = 5$ is the smallest among three cases for $\lambda \leq 1$, while the c.v. of $K = 20$ is the smallest and that of $K = 5$ is the largest for $\lambda \geq 1$. Fig. 2.9 shows that the c.v. becomes large as K increases under LCFS. It is because the small size system has less possibility to find the message with long waiting time than the large size one.

2.6.3 C.V. of Sojourn Time in the System

Figs. 2.10 to 2.12 show c.v.'s of the sojourn time in the system under three service disciplines. In these figures, we can observe that three curves of c.v.'s start from the same value and that the c.v. of FCFS takes the smallest value and that of LCFS takes the largest one among three disciplines. When the arrival rate is small, the sojourn time is almost equal to the service time. When the arrival rate becomes large, the sojourn time is affected by the waiting time.

Let us consider the limiting behavior of the sojourn time in the system when λ tends to infinity. Since the value of the c.v. under LCFS diverges to infinity, we consider FCFS and random scheduling disciplines.

In FCFS, the sojourn time of the tagged message is almost equal to the sum of service time of $K-1$ other messages and that of the tagged one. Hence, we obtain

$$T^{\infty}_{FCFS}(s) \cong \{S^*(s)\}^K.$$

We obtain the first and second moments of the sojourn time as follows.

$$\begin{aligned} T^{\infty(1)}_{FCFS} &= Kb, \\ T^{\infty(2)}_{FCFS} &= K\{(K-1)b^2 + b^{(2)}\}. \end{aligned}$$

Thus, the c.v. becomes

$$C^{\infty}_{T_{FCFS}} = \sqrt{\frac{1}{K}\left(\frac{b^{(2)}}{b^2} - 1\right)}. \tag{2.41}$$

Under random scheduling, the sojourn time is the sum of the remaining service time (almost equal to $S^*(s)$) and the service time of i messages with probability

$$\frac{1}{K-1}\left(\frac{K-2}{K-1}\right)^{i-1}.$$

Thus, the LST of the sojourn time becomes

$$\begin{aligned} T^{\infty}_{RANDOM}(s) &= \sum_{i=1}^{\infty}\{S^*(s)\}^2 \cdot \frac{1}{K-1}\left(\frac{K-2}{K-1}S^*(s)\right)^{i-1} \\ &= \frac{\{S^*(s)\}^2}{K-1-(K-2)S^*(s)}. \end{aligned}$$

First and second moments of T^{∞}_{RANDOM} are expressed as

$$\begin{aligned} T^{\infty(1)}_{RANDOM} &= Kb, \\ T^{\infty(2)}_{RANDOM} &= Kb^{(2)} + 2\{(K-1)b\}^2. \end{aligned}$$

Hence, we obtain the c.v. of the sojourn time as

$$C^{\infty}_{T_{RANDOM}} = \frac{\sqrt{Kb^{(2)} + (K^2-4K+2)b^2}}{Kb}. \tag{2.42}$$

Using above results, we calculate limit values of the two cases (Table 2.1). We can observe that

	Exponential	Erlangian	Hyper-exp.
$C^{\infty}_{T_{FCFS}}$	0.31623	0.18257	0.48304
$C^{\infty}_{T_{RANDOM}}$	0.90554	0.86795	0.97639

Table 2.1: Limit Values of C.V.

c.v.'s under FCFS and LCFS tend to values in Table 2.1.

2.7 Conclusion

This chapter considers the waiting time of the $M/G/1/K$ system under random scheduling and LCFS. Using the analytical results, we derived the LSTs of the waiting time distribution under two service disciplines. We calculated the mean and the coefficient of variation of the waiting time and the sojourn time in the system. Comparing those values under three service disciplines, we showed the influence of the service discipline on the waiting time. We also considered the limiting behavior of the sojourn time under FCFS and random scheduling.

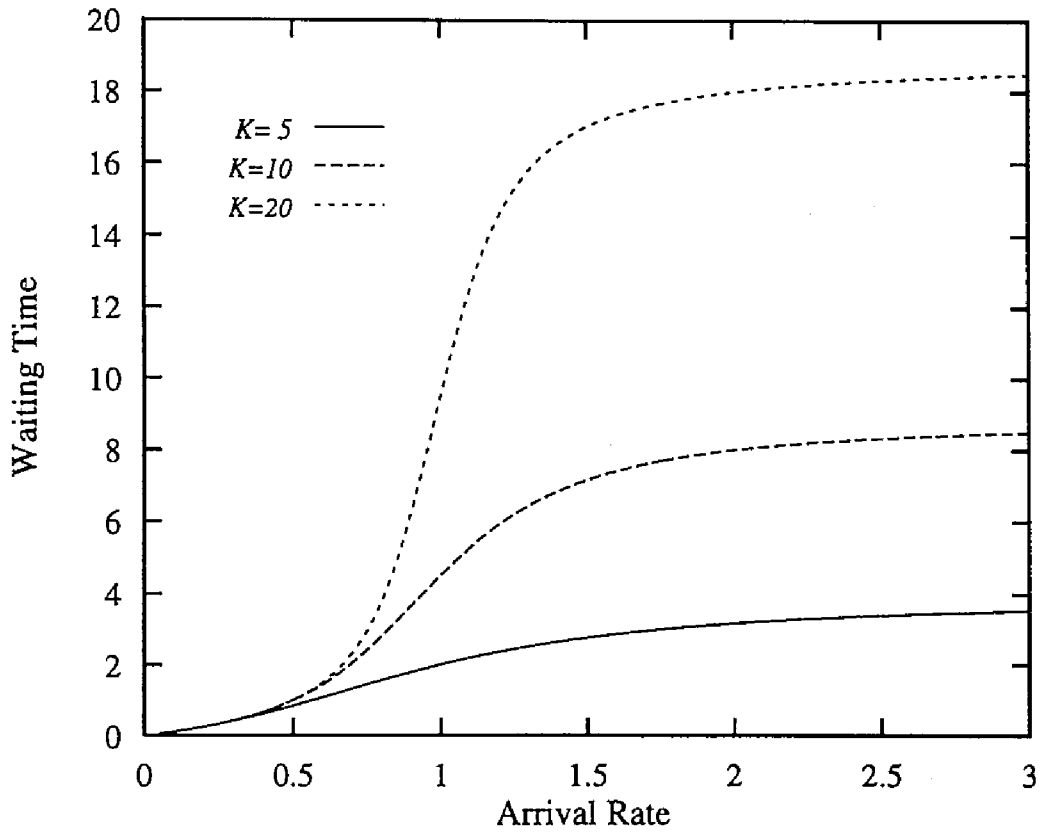


Figure 2.1: Mean Waiting Time ($k = 1$)

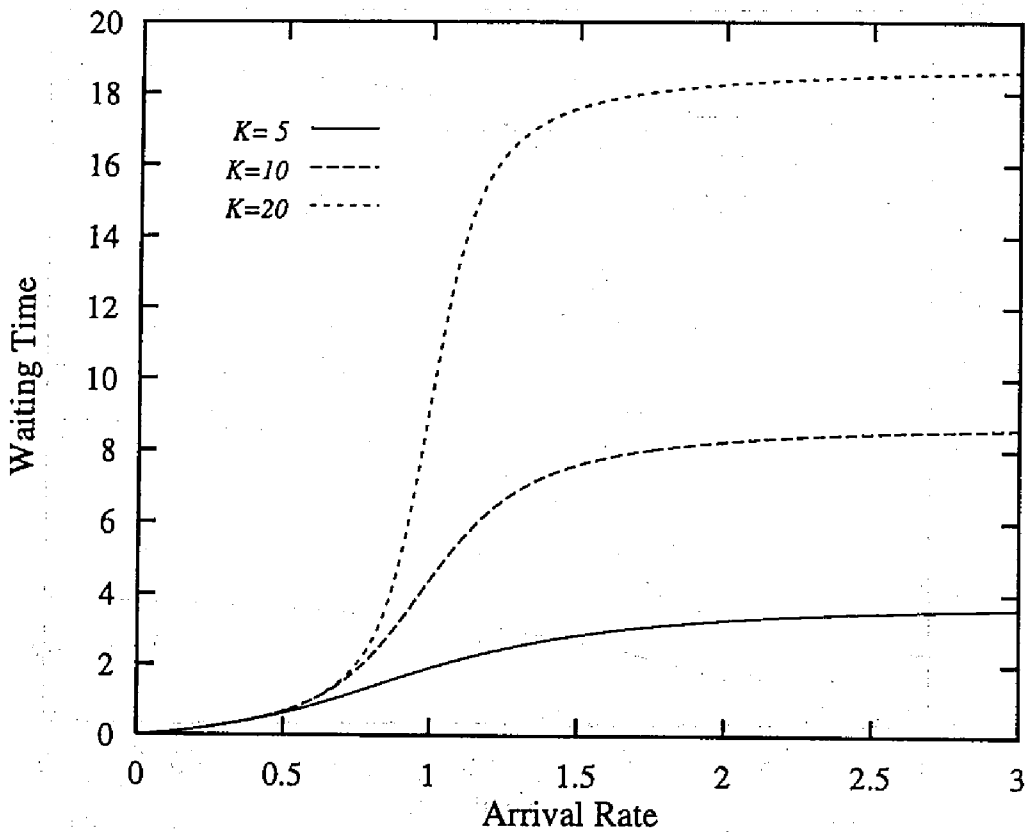


Figure 2.2: Mean Waiting Time ($k = 3$)

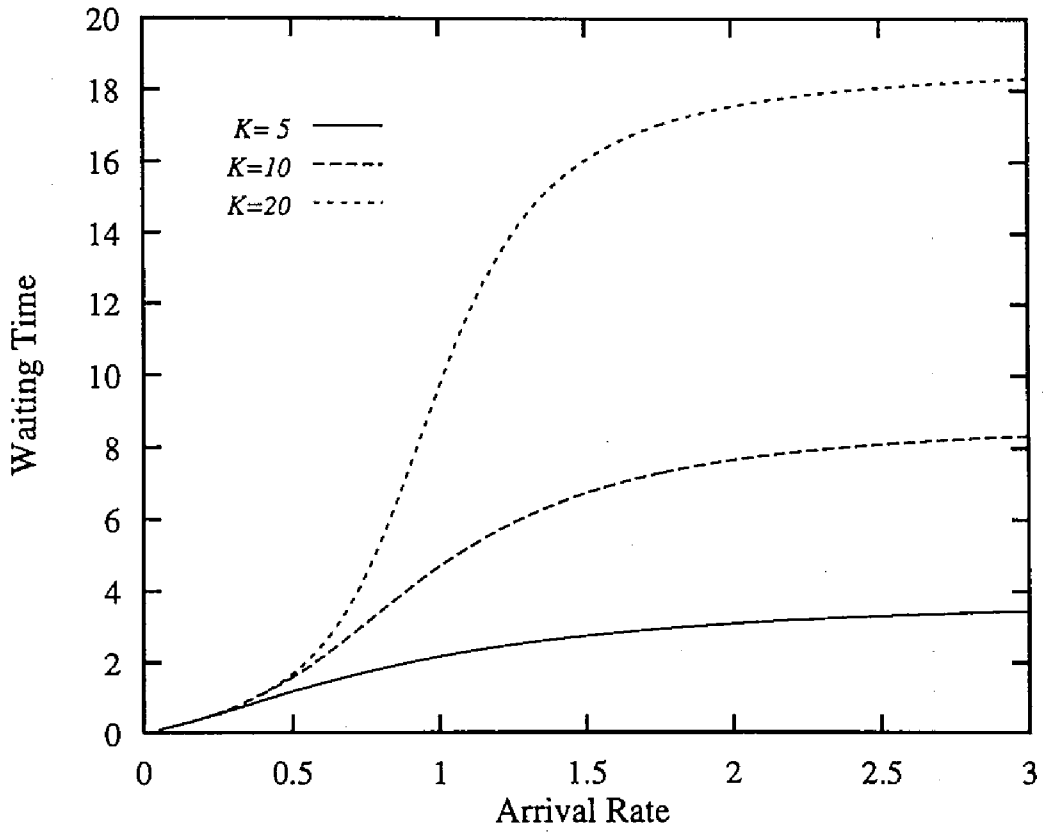


Figure 2.3: Mean Waiting Time (Hyper-exponential)

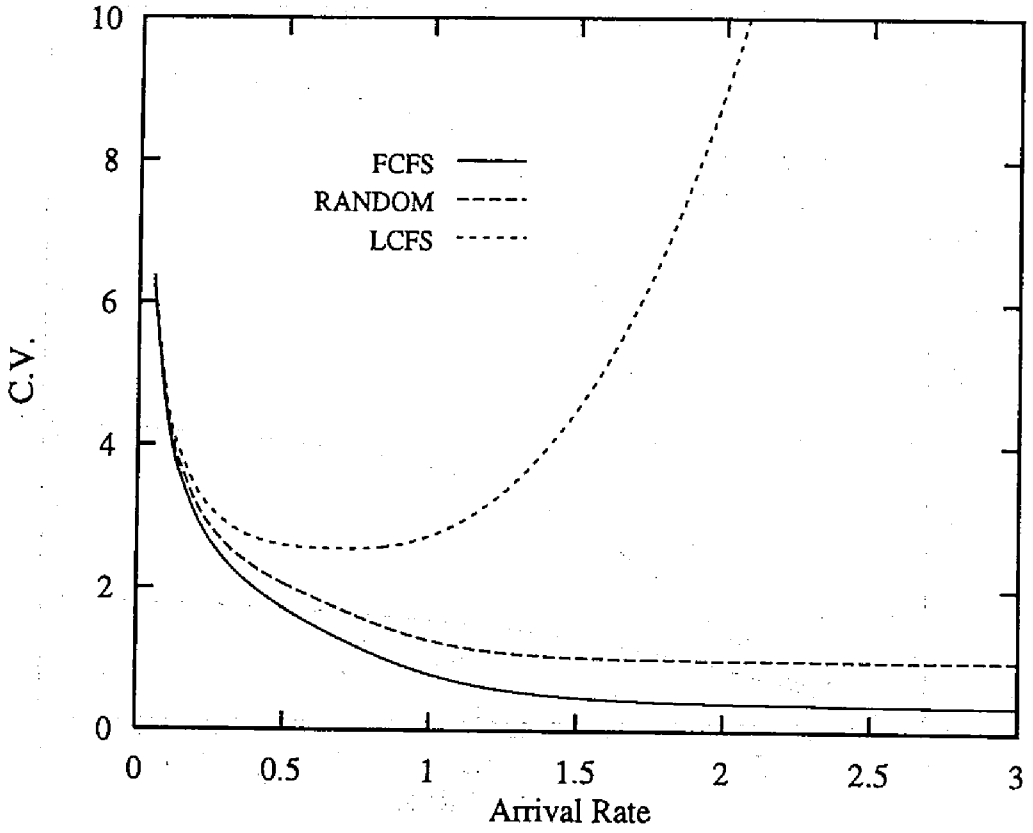


Figure 2.4: C.V. under Three Service Disciplines ($K = 10$, $k = 1$)

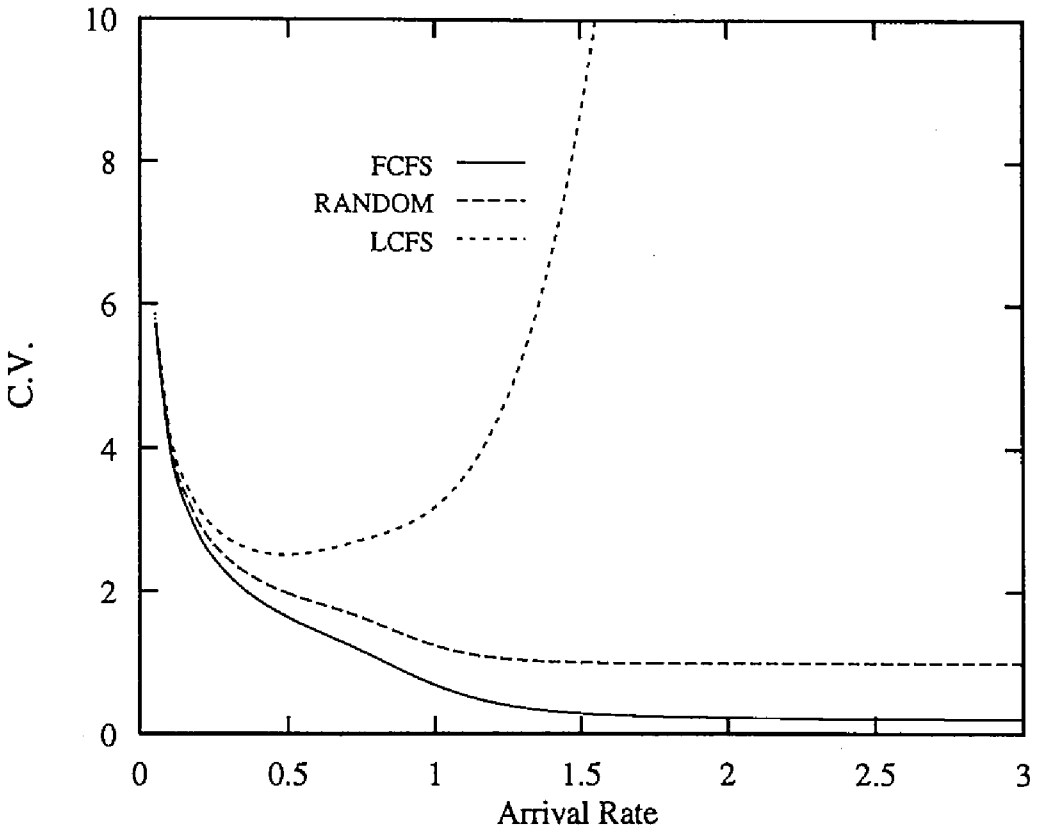


Figure 2.5: C.V. under Three Service Disciplines ($K = 10$, $k = 3$)

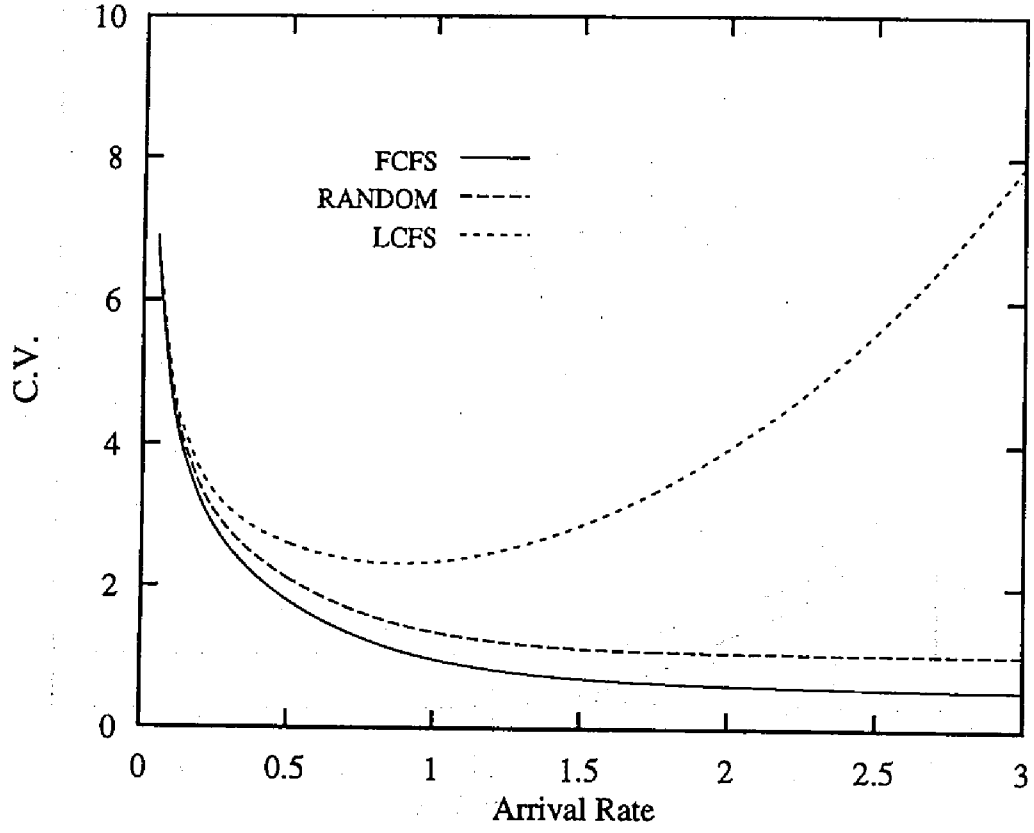


Figure 2.6: C.V. under Three Service Disciplines ($K = 10$, Hyper-exponential)

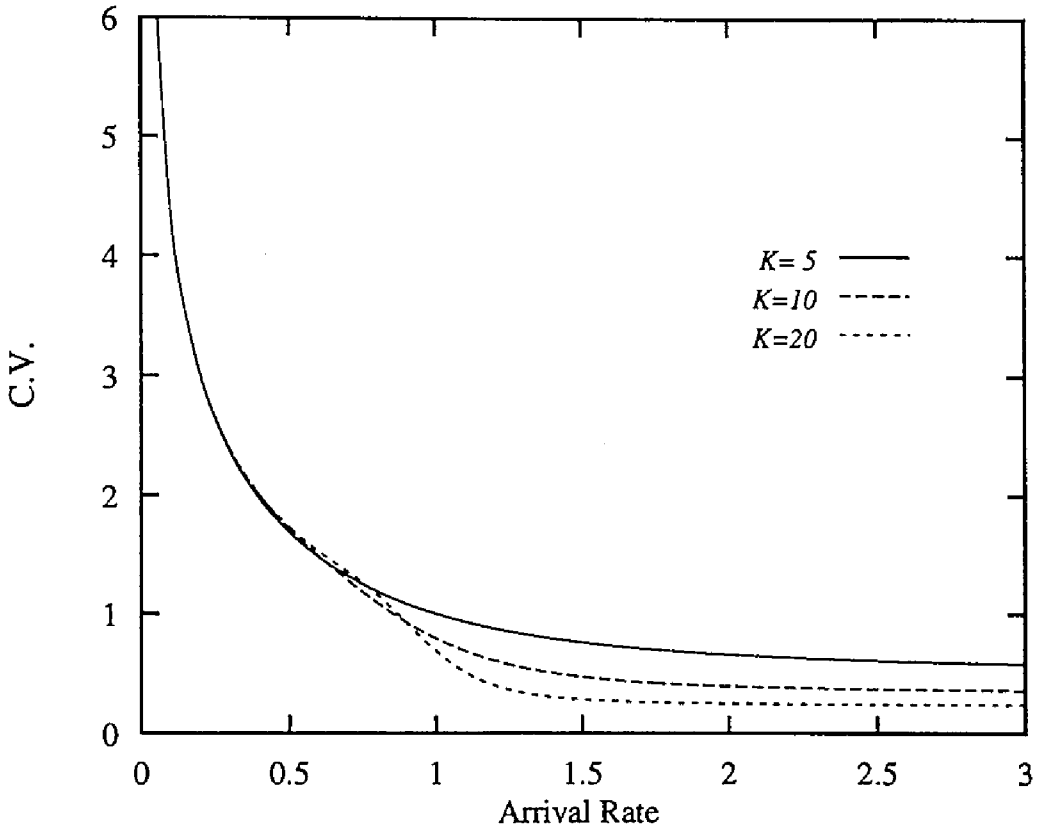


Figure 2.7: C.V. under FCFS ($k = 1$)

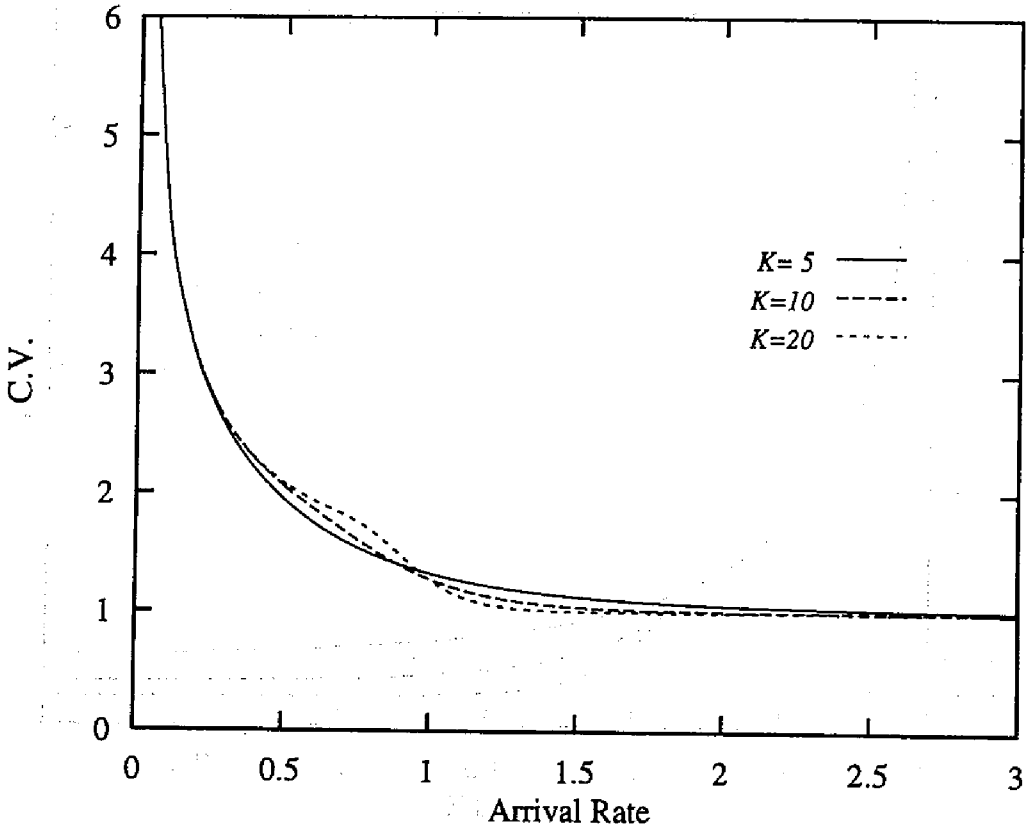


Figure 2.8: C.V. under Random Scheduling ($k = 1$)

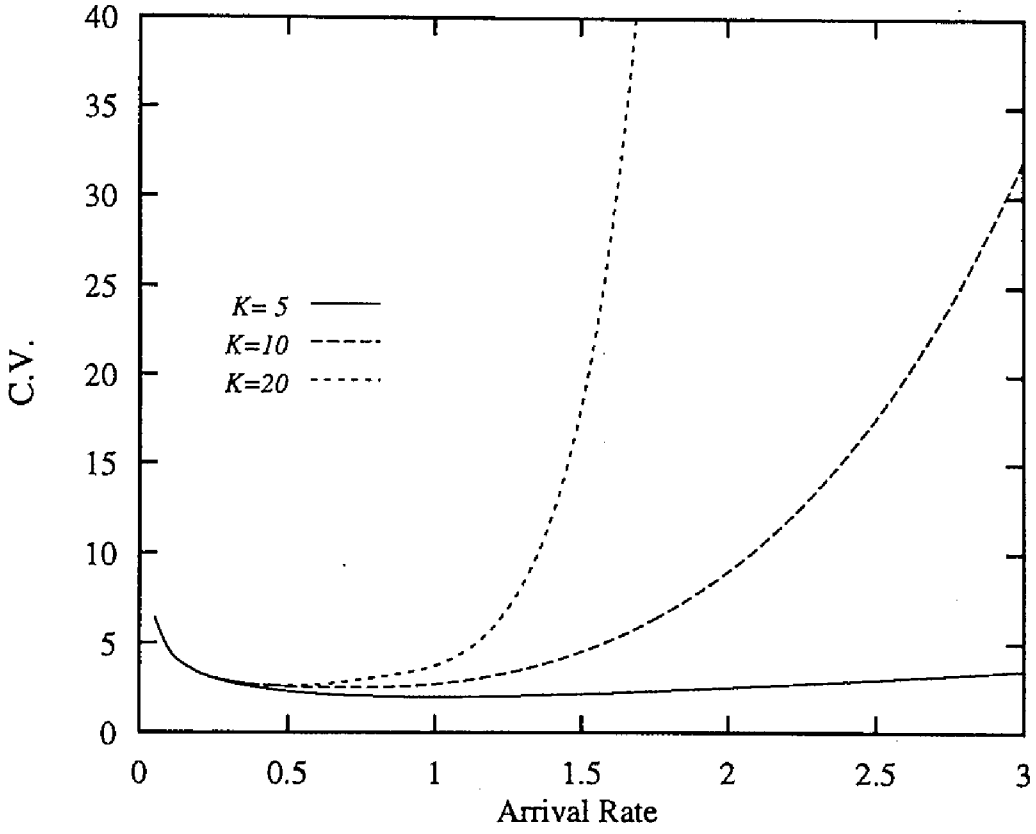


Figure 2.9: C.V. under LCFS ($k = 1$)

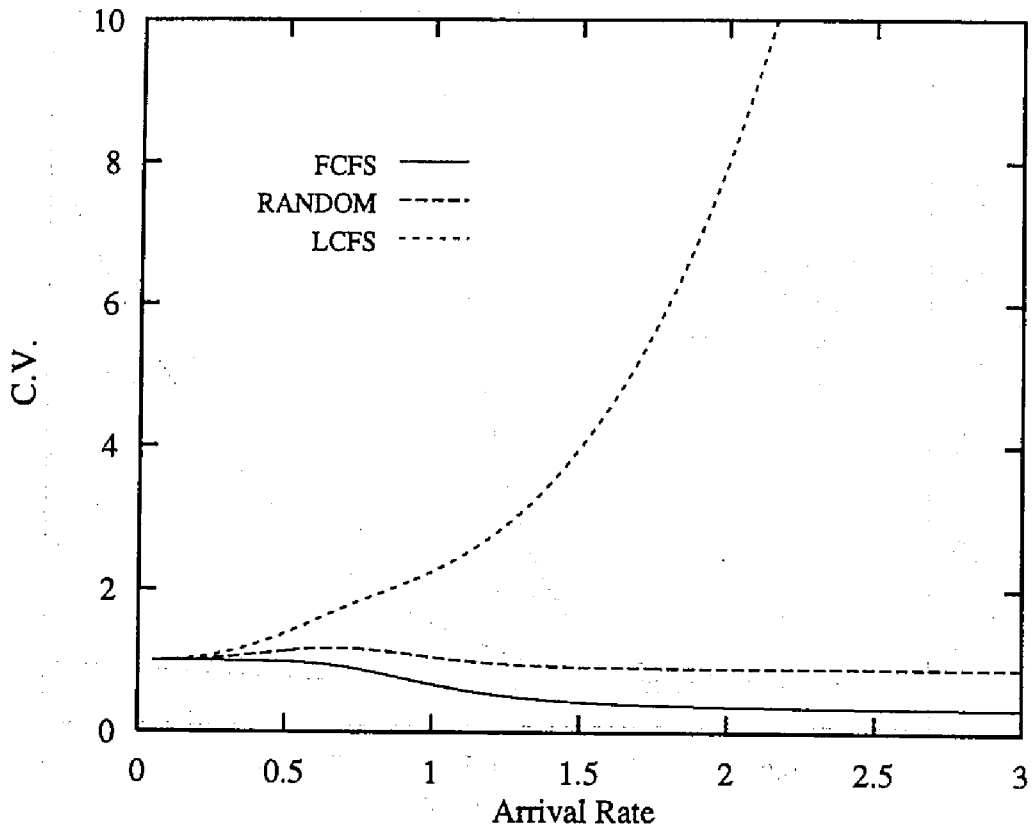


Figure 2.10: C.V. of the Sojourn Time under Three Service Disciplines ($K = 10$, $k = 1$)

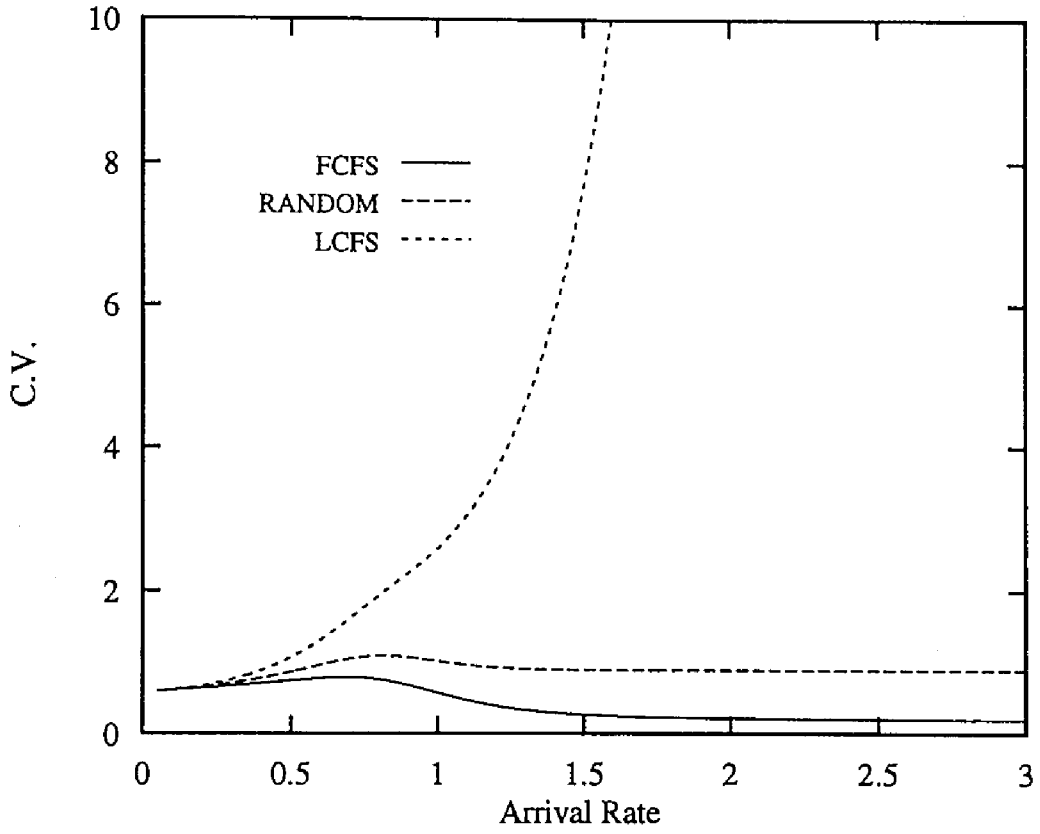


Figure 2.11: C.V. of the Sojourn Time under Three Service Disciplines ($K = 10$, $k = 3$)

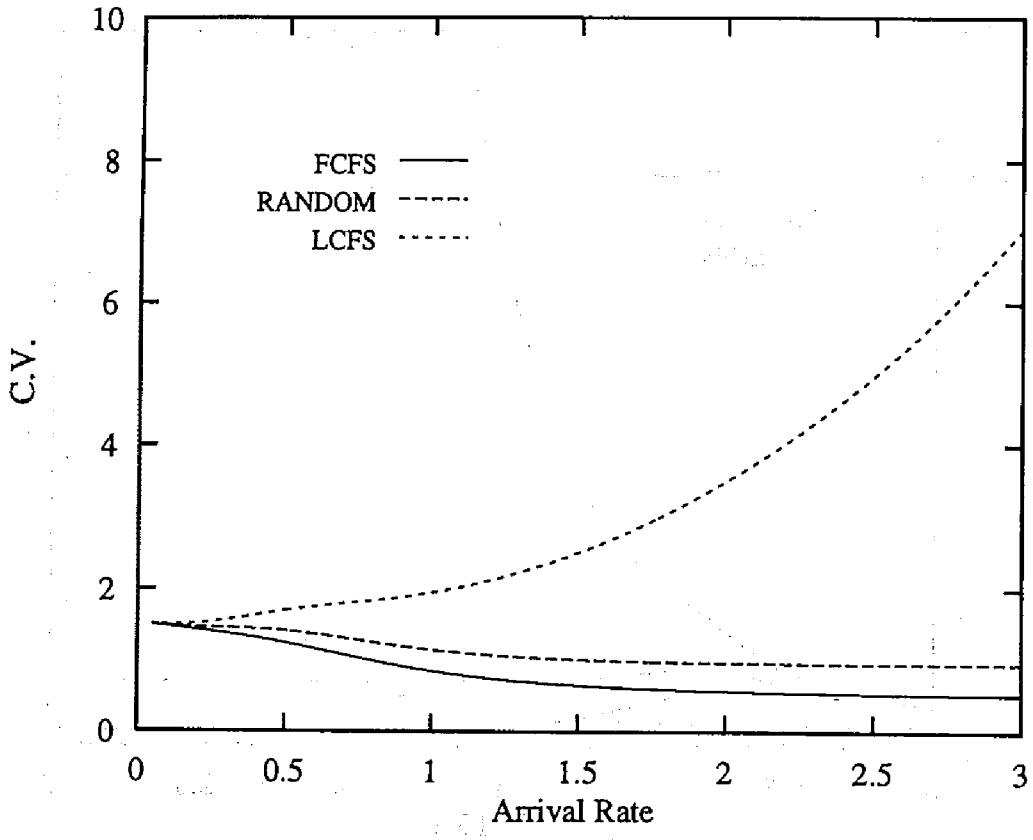


Figure 2.12: C.V. of the Sojourn Time under Three Service Disciplines ($K = 10$, Hyper-exponential)

Chapter 3

M/G/1/K with Vacations under Random Scheduling and LCFS

3.1 Introduction

In this chapter, we analyze the waiting time of an $M/G/1/K$ system with server vacation under random scheduling and LCFS. The subject in this chapter is to compare the performance measures under above three service disciplines and to inspect the influence of the vacation. We derive the LSTs of the waiting time distribution under these disciplines in the similar manner to that in Chapter 2. For the calculation of the performance measures, we present the numerical procedures in detail. Then, we show some numerical results under several conditions. We also analyze the limiting behavior of the system considering the c.v. of the waiting time.

We explain the model of an $M/G/1/K$ system with multiple vacations in section 3.2. In section 3.3, we show the Lee's results of the queue length, and the joint distribution of the number of messages and of the remaining service or vacation time at an arbitrary instant. In section 3.4, we derive the distribution functions of the message waiting time under the two service disciplines. We explain the calculation method and show some numerical results in section 3.5.

3.2 Model

We consider an $M/G/1$ system with finite capacity. Messages arrive at the system according to a Poisson process with a parameter λ. The PDF and the mean of the service time for a message are denoted by $S(x)$ and b, respectively. The vacation policy of our model is multiple vacations, i.e. the server takes vacations repeatedly until he finds at least one waiting message accommodated in upon returning from a vacation. Let $V(x)$ be the PDF for the length of a vacation. The maximum number of messages that can be present in the system is $K < \infty$. When K messages are in the system, new arriving messages cannot enter the system.

3.3 Queue Length Distribution

In this section, we consider the number of messages in the system by the method of imbedded Markov chains [Coop81, Klei75, Taka89].

We choose a set of imbedded Markov points at those points in time when a service is completed or when a vacation ends. Let L_n be the number of messages in the system immediately

after the nth Markov point, and let

$$\eta_n = \begin{cases} 0 & \text{if a vacation ends,} \\ 1 & \text{if a service is completed,} \end{cases} \tag{3.1}$$

at the nth Markov point. We consider the limiting probability distributions:

$$\begin{aligned} \omega_k &\equiv \lim_{n\to\infty} Prob[\eta_n = 0, L_n = k], && 0 \le k \le K, \\ \pi_k &\equiv \lim_{n\to\infty} Prob[\eta_n = 1, L_n = k], && 0 \le k \le K-1, \end{aligned}$$

which satisfy the following equations:

$$\begin{aligned} \omega_k &= (\omega_0 + \pi_0) f_k, && 0 \le k \le K-1, \\ \omega_K &= (\omega_0 + \pi_0) \sum_{m=K}^{\infty} f_m, \\ \pi_k &= \sum_{j=1}^{k+1} (\omega_j + \pi_j) a_{k-j+1}, && 0 \le k \le K-2, \\ \pi_{K-1} &= \omega_K + \sum_{j=1}^{K-1} (\omega_j + \pi_j) \sum_{m=K-j}^{\infty} a_m, \end{aligned} \tag{3.2}$$

and

$$\sum_{k=0}^{K} \omega_k + \sum_{k=0}^{K-1} \pi_k = 1, \tag{3.3}$$

where

$$a_k \equiv \int_0^{\infty} \frac{(\lambda x)^k}{k!} e^{-\lambda x} dS(x), \qquad k = 0, 1, 2, \cdots, \tag{3.4}$$

and

$$f_k \equiv \int_0^{\infty} \frac{(\lambda x)^k}{k!} e^{-\lambda x} dV(x), \qquad k = 0, 1, 2, \cdots. \tag{3.5}$$

From (3.2) and (3.3), we can obtain $\omega_k (k = 0, \cdots, K)$ and $\pi_k (k = 0, \cdots, K-1)$.

Next, we will find the loss probability P_B and the throughput γ of the system. Let us first note from (3.2) that

$$\omega_0 + \pi_0 = \sum_{k=0}^{K} \omega_k, \tag{3.6}$$

is the probability that an arbitrary Markov point is a vacation termination point. Therefore, $1-\omega_0-\pi_0$ is the probability that an arbitrary Markov point is a service completion point. Let ρ be the ratio of the mean service time to the mean interarrival time, and ρ' the server utilization. Let us denote by the reciprocal of σ the mean length of the interval between consecutive imbedded points. It is given by

$$\sigma^{-1} = (\omega_0 + \pi_0) E[V] + (1 - \omega_0 - \pi_0) b. \tag{3.7}$$

From the theorem on the limiting probabilities of semi-Markov process, we obtain

$$\rho' = \frac{(1 - \omega_0 - \pi_0) b}{(\omega_0 + \pi_0) E[V] + (1 - \omega_0 - \pi_0) b} = \sigma (1 - \omega_0 - \pi_0) b, \tag{3.8}$$

and

$$\omega_0 + \pi_0 = \frac{1 - \rho'}{\sigma E[V]} = 1 - \frac{\rho'}{\sigma b}. \tag{3.9}$$

In terms of ρ', σ and $\omega_0 + \pi_0$, the loss probability P_B and the throughput γ of the system are given by

$$P_B = 1 - \frac{\rho'}{\rho}, \tag{3.10}$$

$$\gamma = \lambda(1 - P_B) = \sigma(1 - \omega_0 - \pi_0). \tag{3.11}$$

Next, we consider the joint distribution of the state ξ of the server, the number L of messages in the system, and the remaining vacation time $\widetilde{V}$ when the server is on vacation or the remaining service time $\widetilde{S}$ when the server is busy at an arbitrary instant [Lee84, Taka89]. The state ξ of the server is defined as

$$\xi = \begin{cases} 0 & \text{the server is on vacation,} \\ 1 & \text{the server is busy.} \end{cases} \tag{3.12}$$

We also define

$$\begin{aligned} \Omega_k^*(s) &\equiv \int_0^\infty e^{-sy} Prob[\xi = 0, L = k, y < \widetilde{V} < y + dy], \qquad 0 \le k \le K, \\ \Pi_k^*(s) &\equiv \int_0^\infty e^{-sy} Prob[\xi = 1, L = k, y < \widetilde{S} < y + dy], \qquad 1 \le k \le K. \end{aligned}$$

and $\Omega_k(x)$ and $\Pi_k(x)$ are the inverse transforms of LST $\Omega_k^*(s)$ and $\Pi_k^*(s)$, respectively. Let $\alpha(X)$ be the number of arrivals during the period of length X. Then, we suppose the server is busy and that k messages are in the system. In this case, some (say $j(\ge 1)$) of them were already in the system at the last imbedded point and the rest of them arrived during the elapsed service time $\widehat{S}$. Then, we obtain

$$\begin{aligned} \Pi_k^*(s) &= \frac{\rho'}{1 - \omega_0 - \pi_0} \sum_{j=1}^{k} (\omega_j + \pi_j) E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = k - j] \\ &\qquad \cdot Prob[\alpha(\widehat{S}) = k - j], \quad 1 \le k \le K - 1, \end{aligned} \tag{3.13}$$

$$\begin{aligned} \Pi_K^*(s) &= \frac{\rho'}{1 - \omega_0 - \pi_0} \left\{ \sum_{j=1}^{K-1} (\omega_j + \pi_j) \sum_{m=K-j}^{\infty} E[e^{-s\widetilde{S}} | \alpha(\widehat{S}) = m] \right. \\ &\qquad \left. \cdot Prob[\alpha(\widehat{S}) = m] + \omega_K E[e^{-s\widetilde{S}}] \right\}. \end{aligned} \tag{3.14}$$

Noting that if there are k messages in the system at an observation instant during a vacation, those messages arrived during the elapsed vacation time $\widehat{V}$. Thus we have

$$\Omega_k^*(s) = (1 - \rho') E[e^{-s\widetilde{V}} | \alpha(\widehat{V}) = k] Prob[\alpha(\widehat{V}) = k], \qquad 0 \le k \le K - 1, \tag{3.15}$$

$$\Omega_K^*(s) = (1 - \rho') \sum_{m=K}^{\infty} E[e^{-s\widetilde{V}} | \alpha(\widehat{V}) = m] Prob[\alpha(\widehat{V}) = m]. \tag{3.16}$$

Using $\alpha_n(s)$ of (2.24) in Chapter 2, we write (3.13) as

$$\begin{aligned} \Pi_k^*(s) &= \frac{\rho'}{1 - \omega_0 - \pi_0} \sum_{j=1}^{k} (\omega_j + \pi_j) \frac{1}{\rho} \\ &\qquad \cdot \left[S^*(s) \left(\frac{\lambda}{\lambda - s} \right)^{k-j+1} - \sum_{m=0}^{k-j} a_m \left(\frac{\lambda}{\lambda - s} \right)^{k-j-m+1} \right]. \end{aligned} \tag{3.17}$$

However, by using (3.2) we have

$$\sum_{j=1}^{k}(\omega_j+\pi_j)\sum_{m=0}^{k-j}a_m\left(\frac{\lambda}{\lambda-s}\right)^{k-j-m+1}=\sum_{j=0}^{k-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{k-j}. \tag{3.18}$$

Using (3.9) and (3.18) in (3.17), we obtain

$$\Pi_k^*(s) = \frac{\sigma}{\lambda}\left[S^*(s)\sum_{j=1}^{k}(\omega_j+\pi_j)\left(\frac{\lambda}{\lambda-s}\right)^{k-j+1}-\sum_{j=0}^{k-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{k-j}\right], \quad 1\le k\le K-1. \tag{3.19}$$

From (3.14), we also have

$$\Pi_K^*(s) = -\frac{\sigma}{s}\left\{S^*(s)\left[\sum_{j=1}^{K-1}(\omega_j+\pi_j)\left(\frac{\lambda}{\lambda-s}\right)^{K-j}+\omega_K\right]-\sum_{j=0}^{K-1}\pi_j\left(\frac{\lambda}{\lambda-s}\right)^{K-j-1}\right\}. \tag{3.20}$$

In order to calculate $\omega_k^*(s)$ in a similar way as above, we define

$$\varphi_n(s)\equiv E[e^{-s\widetilde{V}}\cdot\frac{(\lambda\widehat{V})^n}{n!}e^{-\lambda\widehat{V}}], \qquad n=0,1,2,\cdots. \tag{3.21}$$

Similarly to (2.24), we obtain

$$\varphi_n(s)=\frac{1}{\lambda E[V]}\left[V^*(s)\left(\frac{\lambda}{\lambda-s}\right)^{n+1}-\sum_{m=0}^{n}f_m\left(\frac{\lambda}{\lambda-s}\right)^{n-m+1}\right]. \tag{3.22}$$

Using (3.2),(3.9) and $\varphi_n(s)$, (3.15) and (3.16) become

$$\Omega_k^*(s) = \frac{\sigma}{\lambda}\left[V^*(s)(\omega_0+\pi_0)\left(\frac{\lambda}{\lambda-s}\right)^{k+1}-\sum_{j=0}^{k}\omega_j\left(\frac{\lambda}{\lambda-s}\right)^{k-j+1}\right], \quad 0\le k\le K-1, \tag{3.23}$$

$$\Omega_K^*(s) = -\frac{\sigma}{s}\left[V^*(s)(\omega_0+\pi 0)\left(\frac{\lambda}{\lambda-s}\right)^{K}-\sum_{j=0}^{K}\omega_j\left(\frac{\lambda}{\lambda-s}\right)^{K-j}\right]. \tag{3.24}$$

3.4 Analysis of Message Waiting Time

We analyze the LST $W^*(s)$ of the waiting time under (1) random scheduling and (2) LCFS. We also present the previous analysis of the mean waiting time under FCFS in Appendix C.2 [Lee84, Lee89a, Takά63].

3.4.1 Random Scheduling

The message waiting time consists of the remaining time to the next imbedded point after arrival and the duration from the imbedded point to the start of its service. To find the LST $W^*(s)$, we use $W_j(x)$ and $W_j^*(s)$ defined in (2.35) to (2.38).

The message waiting time for random scheduling is considered as follows.

1. The server is on vacation.
   If a message finds j messages in the system upon arrival and if k more messages arrive during the remaining vacation time, then the waiting time of the message is the sum of the remaining vacation time and the time whose distribution is given by $W_{j+k+1}(x)$.

2. The server is busy.
   If a message finds j messages in the system upon arrival and if k more messages arrive during the remaining service time, then the waiting time of the message is the sum of the remaining service time and the time whose distribution is given by $W_{j+k}(x)$.

For simplicity, let A be the number of the messages that arrive during the remaining vacation or service time, and we define

$$\begin{aligned}\Omega^*_{j:k}(s) &\equiv \int_0^\infty e^{-sx} Prob[L=j, A=k, x<\widetilde{V}<x+dx],\\ \Pi^*_{j:k}(s) &\equiv \int_0^\infty e^{-sx} Prob[L=j, A=k, x<\widetilde{S}<x+dx].\end{aligned}$$

Then $\Omega^*_{j:k}(s)$ and $\Pi^*_{j:k}(s)$ become

$$\Omega^*_{j:k}(s) = \int_0^\infty \frac{(\lambda x)^k}{k!} e^{-(s+\lambda)x} d\Omega_j(x), \tag{3.25}$$

$$\Pi^*_{j:k}(s) = \int_0^\infty \frac{(\lambda x)^k}{k!} e^{-(s+\lambda)x} d\Pi_j(x), \tag{3.26}$$

where $\Omega_j(x)$ and $\Pi_j(x)$ are the inverse transforms of LST $\Omega^*_j(s)$ and $\Pi^*_j(s)$, respectively. We note that

$$\sum_{k=0}^{\infty} \Omega^*_{j:k}(s) = \Omega^*_j(s), \tag{3.27}$$

$$\sum_{k=0}^{\infty} \Pi^*_{j:k}(s) = \Pi^*_j(s). \tag{3.28}$$

Therefore, we obtain the message waiting time for random service as

$$\begin{aligned}W^*(s) = \frac{1}{1-P_B}\Bigg[&\sum_{j=0}^{K-2}\Bigg\{\sum_{k=0}^{K-j-2}\Omega^*_{j:k}(s)\cdot W^*_{j+k+1}(s)\\ &+\left(\Omega^*_j(s)-\sum_{k=0}^{K-j-2}\Omega^*_{j:k}(s)\right)\cdot W^*_K(s)\Bigg\}+\Omega^*_{K-1}(s)\cdot W^*_K(s)\\ &+\sum_{j=1}^{K-2}\Bigg\{\sum_{k=0}^{K-j-2}\Pi^*_{j:k}(s)\cdot W^*_{j+k}(s)\\ &+\left(\Pi^*_j(s)-\sum_{k=0}^{K-j-2}\Pi^*_{j:k}(s)\right)\cdot W^*_{K-1}(s)\Bigg\}+\Pi^*_{K-1}(s)\cdot W^*_{K-1}(s)\Bigg].\end{aligned} \tag{3.29}$$

3.4.2 LCFS

In either of the cases that the server is on vacation and that the server is busy, the waiting time of the tagged message is the remaining time plus the length of a delayed busy period which starts

with those messages that arrive during the remaining time. Now, we consider a busy period in the case that the server is busy upon arrival of the tagged message. Suppose that the arriving message finds j messages in the system and that there are k new arrivals during the remaining service time. The number of messages left behind in the system at the next imbedded point is $j+k$. The mean length of the busy period initiated by the last arriving message which starts with its service and ends at the beginning of the service of the last but one is $\bar{\theta}_{K-j-k+1}$ where $\bar{\theta}_K$ is defined in (2.29) to (2.32). Similarly, the mean length of the busy period of the last but two is $\bar{\theta}_{K-j-k+2}$, and so on. After these periods, the service of the tagged message starts.

When the tagged message arrives at the system while the server is busy, one of the following cases arises.

1. The tagged message finds $j(\leq K-2)$ messages, and during the remaining service time
   (a) $k(0 < k < K-j-1)$ new messages arrive.
   (b) More than $K-j-1$ new messages arrive.
   (c) No message arrives.
2. The tagged message finds $K-1$ messages and new messages arriving during the remaining service time are lost.

The case of vacation is considered similarly. Using $\theta_K^*(s)$ of (2.33), we obtain the LST of the message waiting time as

$$\begin{aligned} W^*(s) = \frac{1}{1-P_B}\Bigg[& \sum_{j=0}^{K-3}\sum_{k=1}^{K-j-2}\Omega_{j:k}^*(s)\cdot\prod_{l=0}^{k-1}\theta_{K-j-l-1}^*(s)+\sum_{j=0}^{K-2}\Omega_{j:0}^*(s) \\ & +\sum_{j=0}^{K-2}\left(\Omega_j^*(s)-\sum_{k=0}^{K-j-2}\Omega_{j:k}^*(s)\right)\cdot\prod_{l=1}^{K-j-1}\theta_l^*(s)+\Omega_{K-1}^*(s) \\ & +\sum_{j=1}^{K-3}\sum_{k=1}^{K-j-2}\Pi_{j:k}^*(s)\cdot\prod_{l=0}^{k-1}\theta_{K-j-l}^*(s)+\sum_{j=1}^{K-2}\Pi_{j:0}^*(s) \\ & +\sum_{j=1}^{K-2}\left(\Pi_j^*(s)-\sum_{k=0}^{K-j-2}\Pi_{j:k}^*(s)\right)\cdot\prod_{l=2}^{K-j}\theta_l^*(s)+\Pi_{K-1}^*(s)\Bigg]. \end{aligned} \tag{3.30}$$

3.5 Numerical Results

We have calculated the mean and the c.v. of the waiting time using the results presented in section 3.4. Before showing numerical examples, we explain the procedure of calculations.

3.5.1 Procedure of Calculations

First of all, we calculate the limiting probability distributions $\{\pi_k; 0 \leq k \leq K-1\}$ and $\{\omega_k; 0 \leq k \leq K\}$ [Taka89]. We define $\bar{\pi}_k$ as

$$\bar{\pi}_k = \frac{\pi_k+\omega_k}{\pi_0+\omega_0}, \qquad 0 \leq k \leq K-1. \tag{3.31}$$

From (3.2), $\{\bar{\pi}_k : 0 \le k \le K-1\}$ can be recursively calculated by

$$\bar{\pi}_0 = 1, \tag{3.32}$$

$$\bar{\pi}_{k+1} = \frac{1}{a_k}\left(\bar{\pi}_k - \sum_{j=1}^{k}\bar{\pi}_j a_{k-j+1} - f_k\right), \qquad 0 \le k \le K-2. \tag{3.33}$$

From (3.2) and (3.3), we obtain

$$\pi_0 + \omega_0 = \left(\sum_{k=0}^{K-1}\bar{\pi}_k + \sum_{k=K}^{\infty} f_k\right)^{-1} = \left\{1 + \sum_{k=0}^{K-1}(\bar{\pi}_k - f_k)\right\}^{-1}. \tag{3.34}$$

Now, $\{\omega_k : 0 \le k \le K\}$ are obtained from (3.2) and then $\{\pi_k : 0 \le k \le K-1\}$ are calculated from (3.31).

For both random scheduling and LCFS cases, we need to calculate $\Omega^*_{j:k}(0)$, $\Pi^*_{j:k}(0)$ and the first and second derivatives of $\Omega^*_{j:k}(s)$ and $\Pi^*_{j:k}(s)$. Now we show the method of the calculation of $\Pi^*_{j:k}(0)$. Setting $s = 0$ in (3.26), we obtain

$$\Pi^*_{j:k}(0) = \int_0^\infty \frac{(\lambda x)^k}{k!}e^{-\lambda x}d\Pi_j(x) = \frac{(-\lambda)^k}{k!}\left(\frac{d^k}{ds^k}\Pi^*_j(s)\right)_{s=\lambda}. \tag{3.35}$$

We need to derive the k-th derivative of $\Pi^*_j(s)$. Multiplying by $(\lambda - s)^j$ in (3.19) and differentiating $k+j$ times, we obtain

$$\sum_{n=0}^{k+j}\binom{k+j}{n}\frac{d^n}{ds^n}\left\{(\lambda-s)^j\right\}\Pi_j^{*(k+j-n)}(s) = \frac{\sigma}{\lambda}\left[\sum_{i=1}^{j}(\pi_i+\omega_i)\lambda^{j-i+1}\sum_{n=0}^{k+j}\binom{k+j}{n}\frac{d^n}{ds^n}\left\{(\lambda-s)^{i-1}\right\}S^{*(k+j-n)}(s)\right], \tag{3.36}$$

where $U^{*(k)}(s)$ denotes the k-th derivative of $U^*(s)$. Setting $s = \lambda$, (3.36) becomes

$$\frac{1}{k!}\Pi_j^{*(k)}(\lambda) = \frac{\sigma}{\lambda}\sum_{i=1}^{j}(\pi_i+\omega_i)\frac{(-\lambda)^{j-i+1}}{(k+j-i+1)!}S^{*(k+j-i+1)}(\lambda). \tag{3.37}$$

Using (3.35) and (3.37), we obtain

$$\Pi^*_{j:k}(0) = \frac{\sigma}{\lambda}\sum_{i=1}^{j}(\pi_i+\omega_i)\frac{(-\lambda)^{k+j-i+1}}{(k+j-i+1)!}S^{*(k+j-i+1)}(\lambda). \tag{3.38}$$

Using (3.38), the first and the second derivatives of $\Pi^*_{j:k}(0)$ are expressed as

$$\left(\frac{d}{ds}\Pi^*_{j:k}(s)\right)_{s=0} = -\frac{k+1}{\lambda}\Pi^*_{j:k+1}(0), \tag{3.39}$$

$$\left(\frac{d^2}{ds^2}\Pi^*_{j:k}(s)\right)_{s=0} = \frac{(k+2)(k+1)}{\lambda^2}\Pi^*_{j:k+2}(0). \tag{3.40}$$

$\Omega^*_{j:k}(0)$ and its first and second derivatives can be calculated in a similar way.

Under the random scheduling, we need to calculate the derivatives of $W^*_j(s)$ defined in (2.37) and (2.38). When we differentiate (2.37) and (2.38) with respect to s and set $s = 0$, we obtain K linear equations. Thus, derivatives can be calculated by solving those equations.

In LCFS case, derivatives of $\theta^*_l(s)$ can be recursively calculated from (2.33).

After above calculations, the mean and the c.v. of the waiting time under the two cases can be calculated from (3.29) and (3.30).

3.5.2 Numerical Examples

In our numerical examples, the LST of the service time, $S^*(s)$, and that of the vacation time, $V^*(s)$ are chosen as follows.

1. LST of the service time $S^*(s)$ $\cdots$ k phase Erlangian distribution:

$$S^*(s) = \left(\frac{k\mu}{s+k\mu}\right)^k,$$

where $\mu = 1.0$, and $k = 1$(exponential distribution), 2, 5, and 10.

2. LST of the vacation time $V^*(s)$ $\cdots$ exponential distribution:

$$V^*(s) = \frac{v}{s+v},$$

and v takes v =1.0, 2.0, 4.0 and 8.0.

Let the first and second moments of the waiting time be $\bar{W}$ and $W^{(2)}$, respectively. We have calculated the following values ;

1. $\bar{W}$: Mean waiting time

2. C_W : Coefficient of variation of the waiting time

$$C_W = \frac{\sqrt{W^{(2)} - (\bar{W})^2}}{\bar{W}}.$$

We have illustrated the numerical results in Figs.3.1 to 3.12.

Mean Waiting Time

Fig.3.1 illustrates the variation of the mean waiting time for different system sizes. We can observe that the mean waiting time increases suddenly around $\lambda = 1$ and that it approaches a constant value.

The mean waiting time is independent of service disciplines, so we consider the case of FCFS. The number of messages in the system increases according to λ. However, the system size is of a finite value K, and the message that can enter the system sees at most $K-1$ other messages. Thus, using the mean service time b $(= 1/\mu)$, the waiting time of a message is at most $(K-1)b$. In this example, b =1.0, then each value for K =5, 10, 20 and 40 approaches 4, 9, 19 and 39.

Fig.3.2 illustrates the variation of the mean waiting time for different $S^*(s)$'s. When the number of phases increases, the mean waiting time approaches the value for the case of constant service time.

We note that in each graph the mean waiting time never tends to zero even when the arrival rate is quite small. This is because each arriving message is delayed for the remaining vacation time. Since the vacation time is exponentially distributed, the mean remaining vacation time is $1/v$. For $v = 1.0$, the mean waiting time approaches 1 when λ is small.

Coefficient of Variation of Waiting Time

Fig.3.3 illustrates the c.v. under three service disciplines. We can observe that the mean waiting times of FCFS and random scheduling approach finite values, and that of LCFS tends to infinity. When arrival rate λ is small, the waiting time is affected by the remaining vacation time. Let first and second moments of the remaining vacation time $\tilde{V}$ be $\tilde{V}^{(1)}$ and $\tilde{V}^{(2)}$, respectively. Noting that vacation time is exponentially distributed, we obtain the c.v. of the remaining vacation time $\tilde{V}$ as follows:

$$C_{\tilde{V}} = \frac{\sqrt{\tilde{V}^{(2)} - (\tilde{V}^{(1)})^2}}{\tilde{V}^{(1)}} = 1.$$

Thus the value for each discipline starts from 1.

Let us consider the limiting behavior of the waiting time when λ is infinity. Although the c.v. of LCFS becomes infinity, those of FCFS and random order remain finite. In the case, the waiting time of FCFS is almost equal to the sum of service time of $K-1$ other messages. Thus, the LST of the waiting time is

$$W^{*\infty}_{FCFS}(s) = \{S^*(s)\}^{K-1}.$$

Denoting the second moment of $S^*(s)$ by $b^{(2)}$, we obtain the first and second moments of the waiting time as follows:

$$W^{\infty(1)}_{FCFS} = (K-1)b, \tag{3.41}$$

$$W^{\infty(2)}_{FCFS} = (K-1)\{(K-2)b^2 + b^{(2)}\}. \tag{3.42}$$

Thus the coefficient of variation is

$$C^{\infty}_{W_{FCFS}} = \sqrt{\frac{1}{K-1}\left(\frac{b^{(2)}}{b^2} - 1\right)}. \tag{3.43}$$

Under random scheduling and heavy traffic condition, the message waiting time is the sum of the remaining service time(equal to $S^*(s)$) and the service time of i messages with probability

$$\frac{1}{K-1}\left(\frac{K-2}{K-1}\right)^{i-1}.$$

Thus, we have the LST of the waiting time as follows:

$$\begin{aligned} W^{*\infty}_{RANDOM}(s) &= \sum_{i=1}^{\infty} S^*(s) \cdot \frac{1}{K-1}\left(\frac{K-2}{K-1}S^*(s)\right)^{i-1} \\ &= \frac{S^*(s)}{K-1-(K-2)S^*(s)}. \end{aligned} \tag{3.44}$$

First and second moments are expressed as

$$W^{\infty(1)}_{RANDOM} = (K-1)b, \tag{3.45}$$

$$W^{\infty(2)}_{RANDOM} = (K-1)\{b^{(2)} + 2(K-2)b^2\}. \tag{3.46}$$

So the coefficient of variation is

$$C^{\infty}_{W_{RANDOM}} = \frac{\sqrt{(K-1)\{b^{(2)} + (K-3)b^2\}}}{(K-1)b}. \tag{3.47}$$

Using above results, we can calculate the limit values of the two cases as

$$\begin{aligned} C^{\infty}_{W_{FCFS}} &= \frac{1}{3}, \\ C^{\infty}_{W_{RANDOM}} &= 1. \end{aligned}$$

We can observe that two curves in Fig.3.3 approach these values.

Figs.3.4 to 3.6 illustrate the numerical results under three service disciplines while the mean vacation time changes. When the vacation rate is large, the c.v. is large. This is because the mean remaining vacation time decreases as the mean vacation time $1/v$ does. To be more concrete, we set $\mu = 1.0$ and compare the two cases of $v = 1.0$ and 8.0. If the tagged message finds the system empty, the mean waiting time $\bar{W}$ is

$$\bar{W}[L=0] = \begin{cases} 1.0 & \text{if } v = 1.0 \\ \frac{1}{8} & \text{if } v = 8.0. \end{cases}$$

If the message finds one message in the system, then the mean waiting time is

$$\bar{W}[L=1] = \begin{cases} 2.0 & \text{if } v = 1.0 \\ \frac{9}{8} & \text{if } v = 8.0. \end{cases}$$

In light traffic, the probability of multiple messages in the system is small. Thus the variation of the waiting time under $v = 8.0$ is larger than that under $v = 1.0$. So the c.v. becomes large when the value of v is large.

Figs.3.7 to 3.9 illustrate the numerical results of the c.v. in the case that the phase of Erlangian distribution is changed. As the number of the phases increases, the values under FCFS and random scheduling become small and that under LCFS becomes large.

In FCFS, the mean waiting time varies only slightly when the number of the phases is large. This is simply because the service time tends to constant. On the other hand, under random scheduling c.v. remains stable, because random selection for service still fluctuates the waiting time.

Figs.3.10 to 3.12 illustrate the behavior of the c.v. in the case that the system size K is changed. We can observe that under FCFS and random order, the curves greatly vary around $\lambda = 1$, and that the value of the c.v. under LCFS is large when K is large. Under FCFS, the variation of the waiting time becomes large in proportion to K. On the other hand, in random scheduling, increasing K affects the probability of selection for service. In LCFS, the messages in the system become hard to be served when K becomes large.

3.6 Conclusion

In this chapter, we analyzed the waiting time of the $M/G/1/K$ system with server vacations under random scheduling and LCFS. Using the analytical results, we derived the LSTs of the waiting time. We also computed the mean and the coefficient of variation of the waiting time and compared those values under three service disciplines. From the numerical results, we found that the waiting time is influenced by the remaining vacation time and the selection of service disciplines.

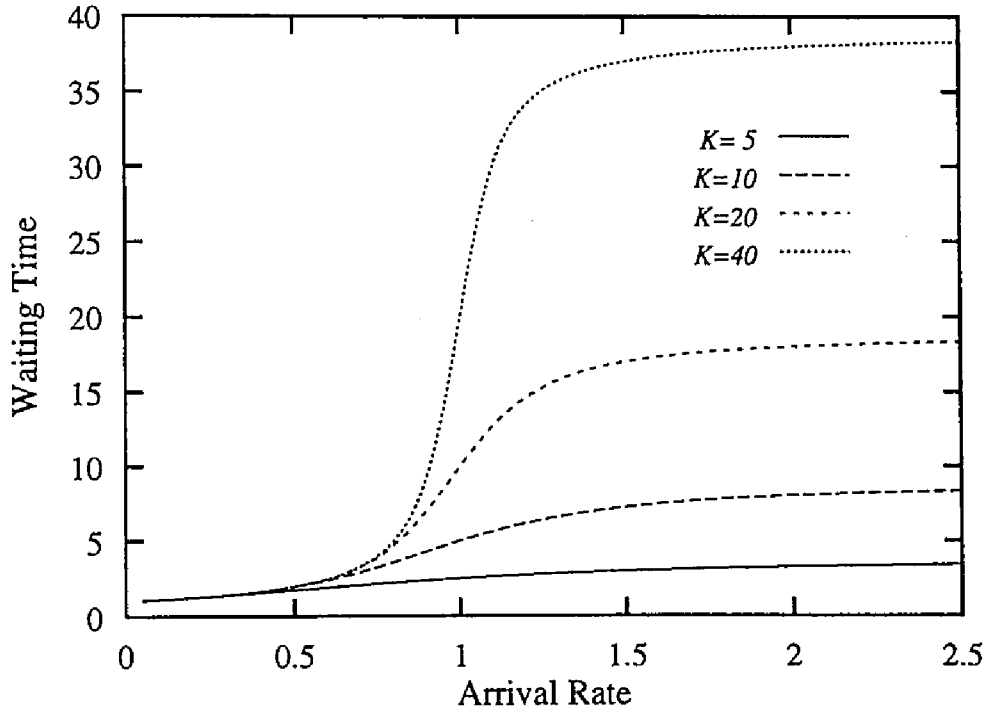


Figure 3.1: Mean Waiting Time ($k = 1$, $v = 1$)

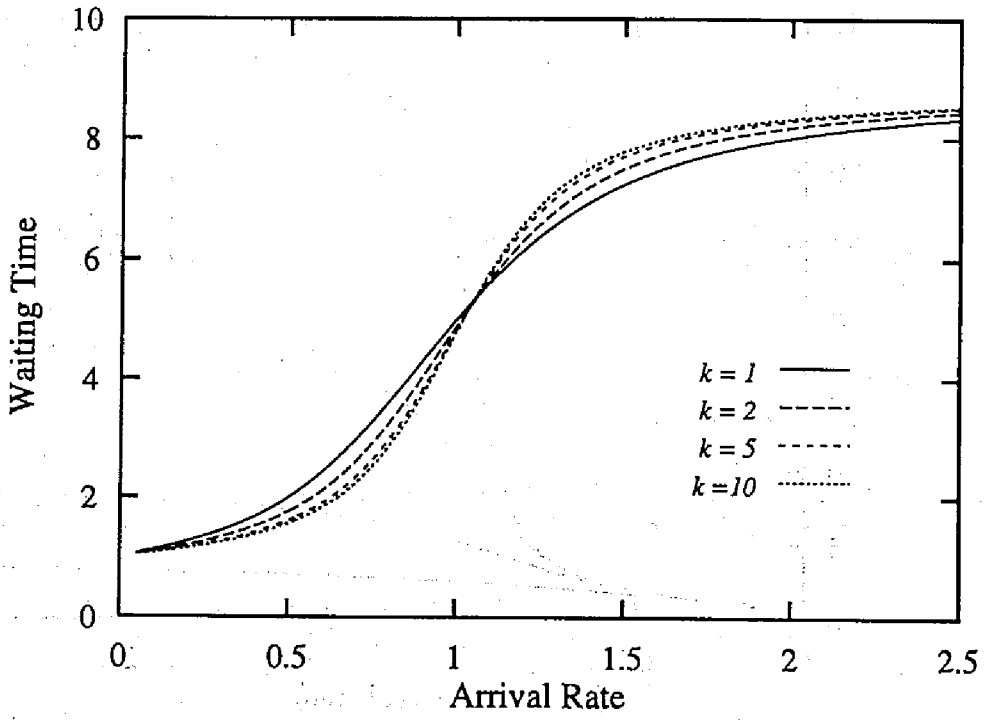


Figure 3.2: Mean Waiting Time ($K = 10$, $v = 1$)

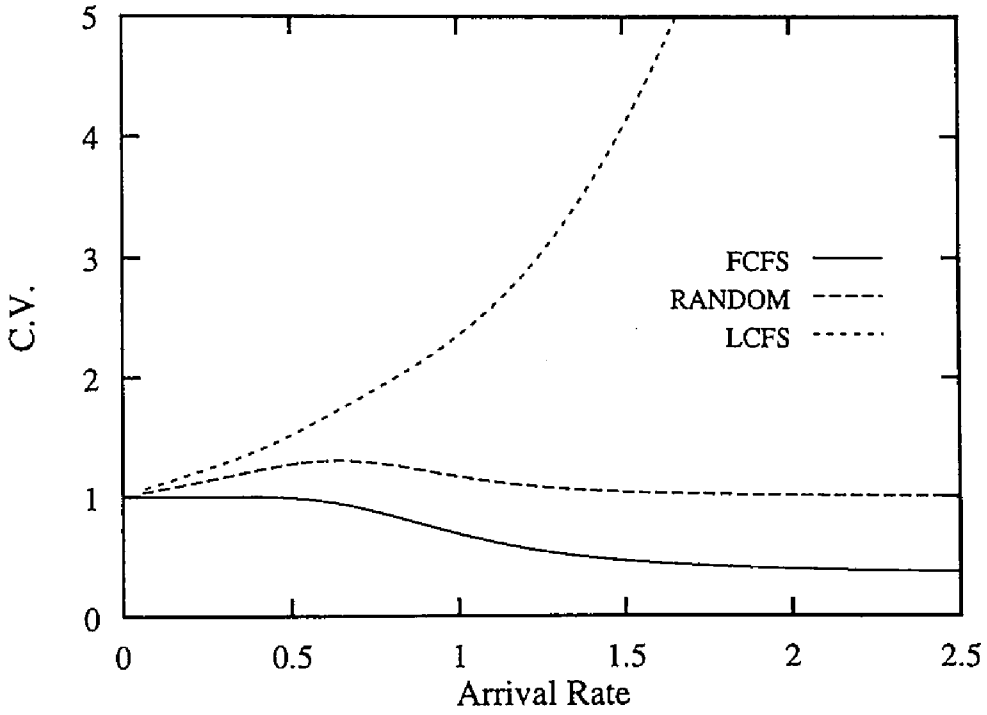


Figure 3.3: C.V. under Three Service Disciplines ($K = 10$, $k = 1$, $v = 1$)

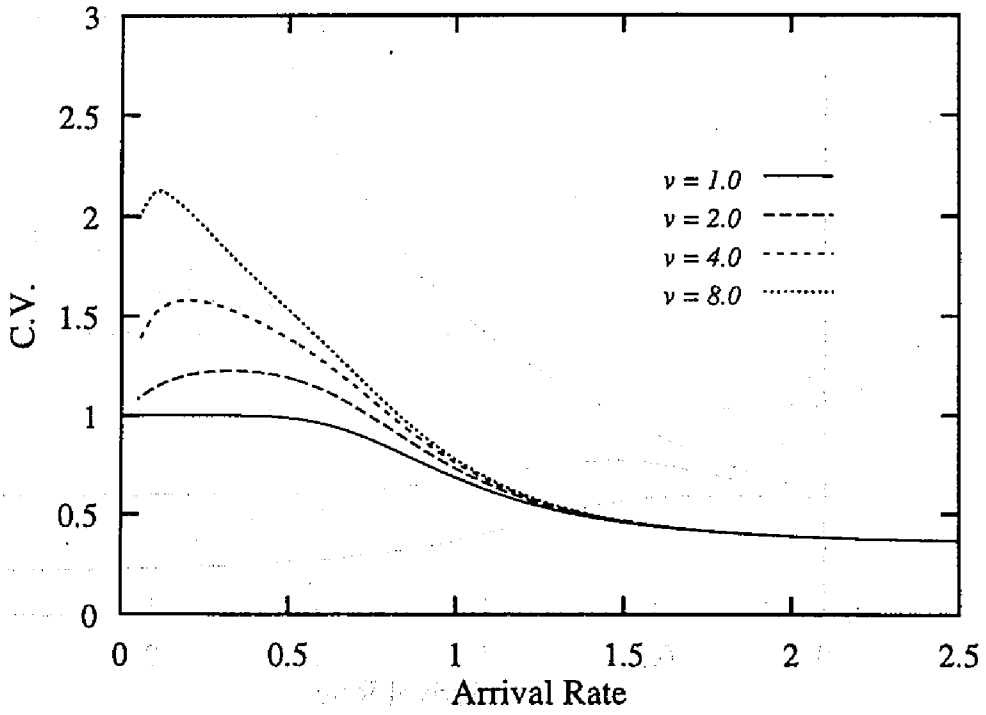


Figure 3.4: C.V. under FCFS ($K = 10$, $k = 1$)

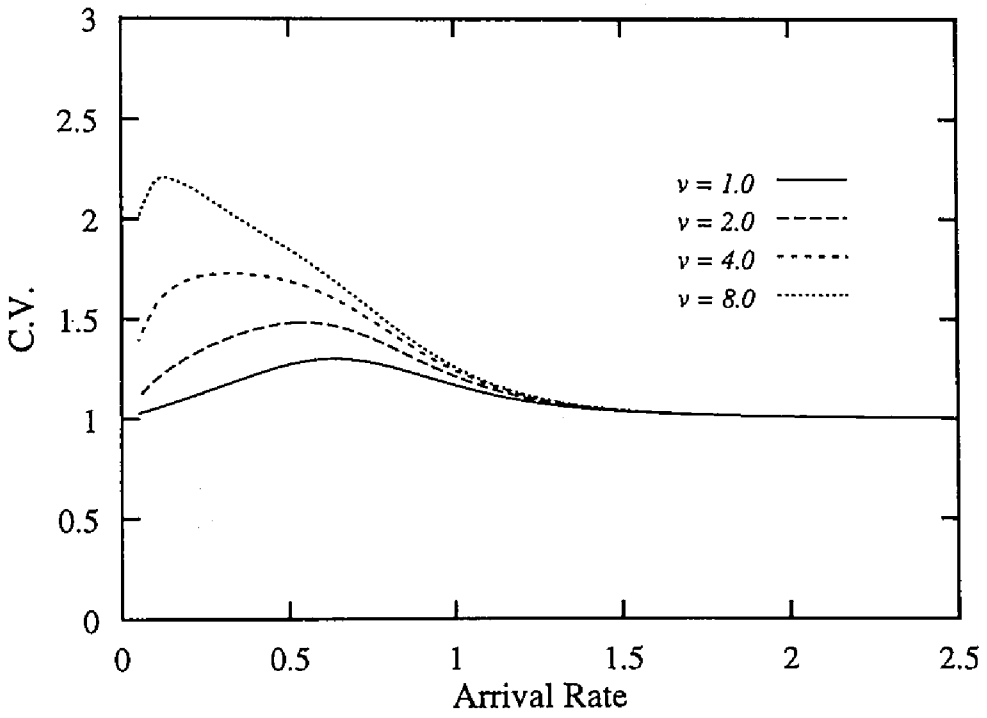


Figure 3.5: C.V. under Random Scheduling ($K = 10$, $k = 1$)

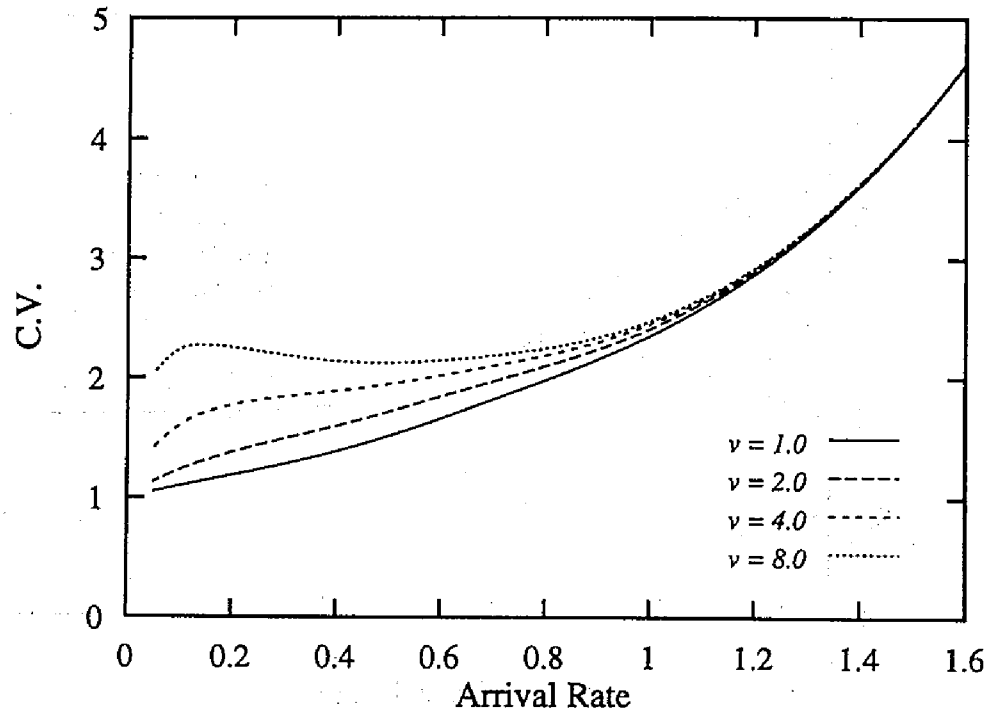


Figure 3.6: C.V. under LCFS ($K = 10$, $k = 1$)

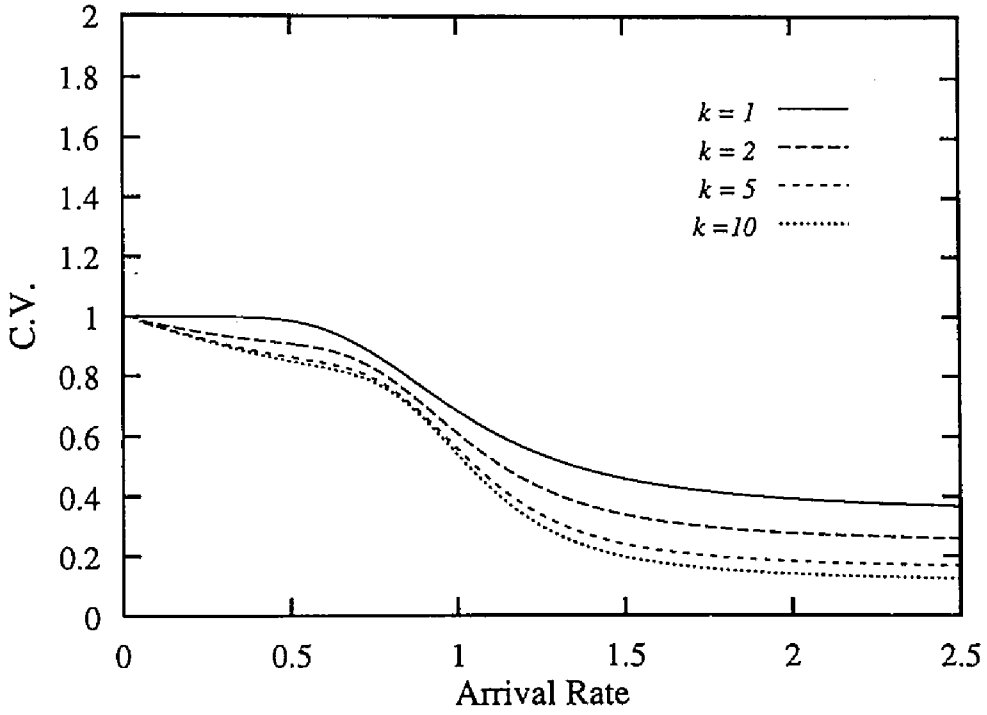


Figure 3.7: C.V. under FCFS ($K = 10$, $v = 1$)

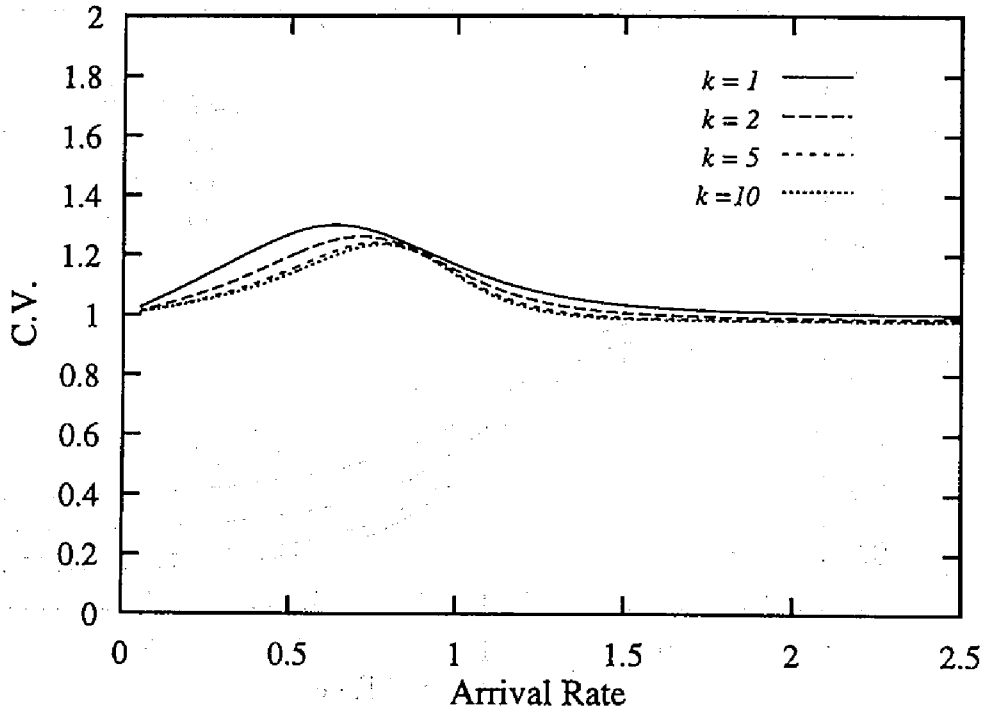


Figure 3.8: C.V. under Random Scheduling ($K = 10$, $v = 1$)

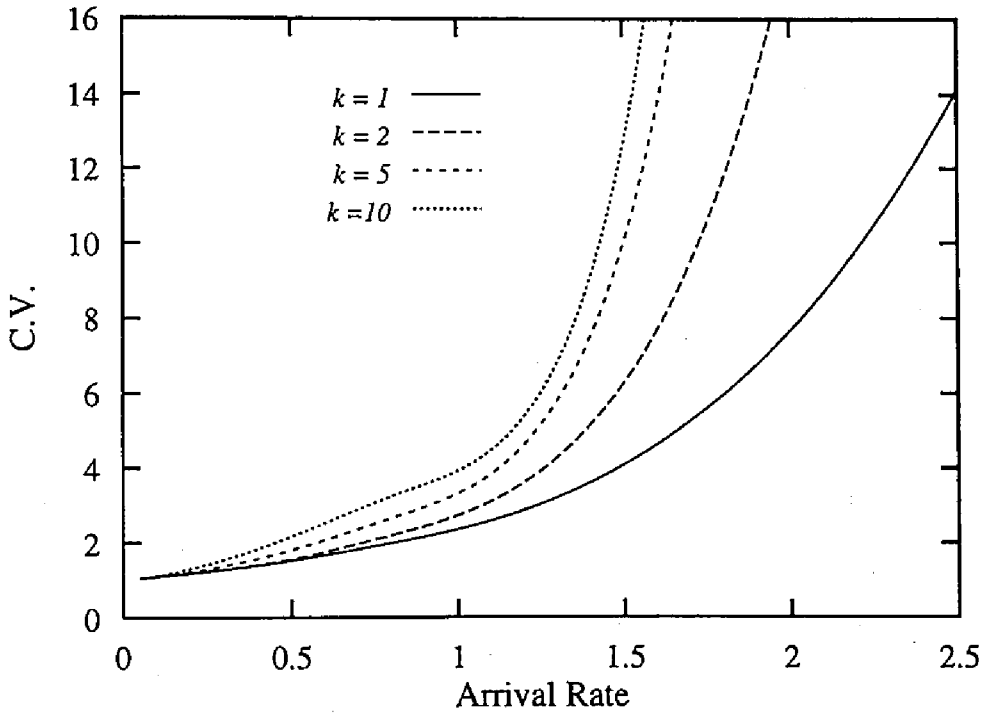


Figure 3.9: C.V. under LCFS ($K = 10, v = 1$)

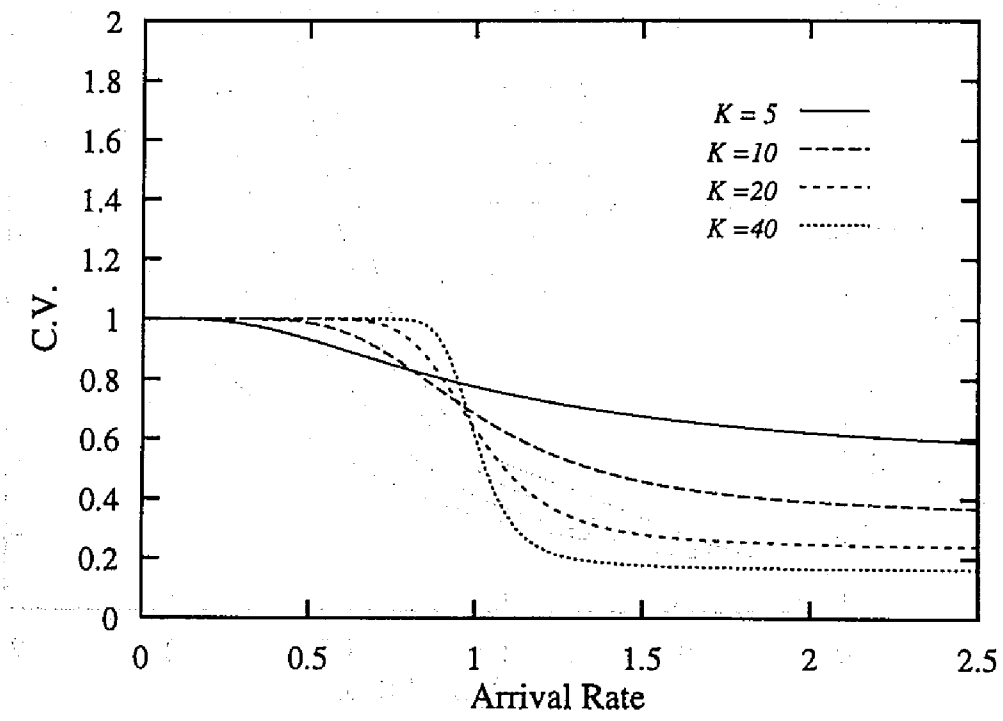


Figure 3.10: C.V. under FCFS ($k = 1$, $v = 1$)

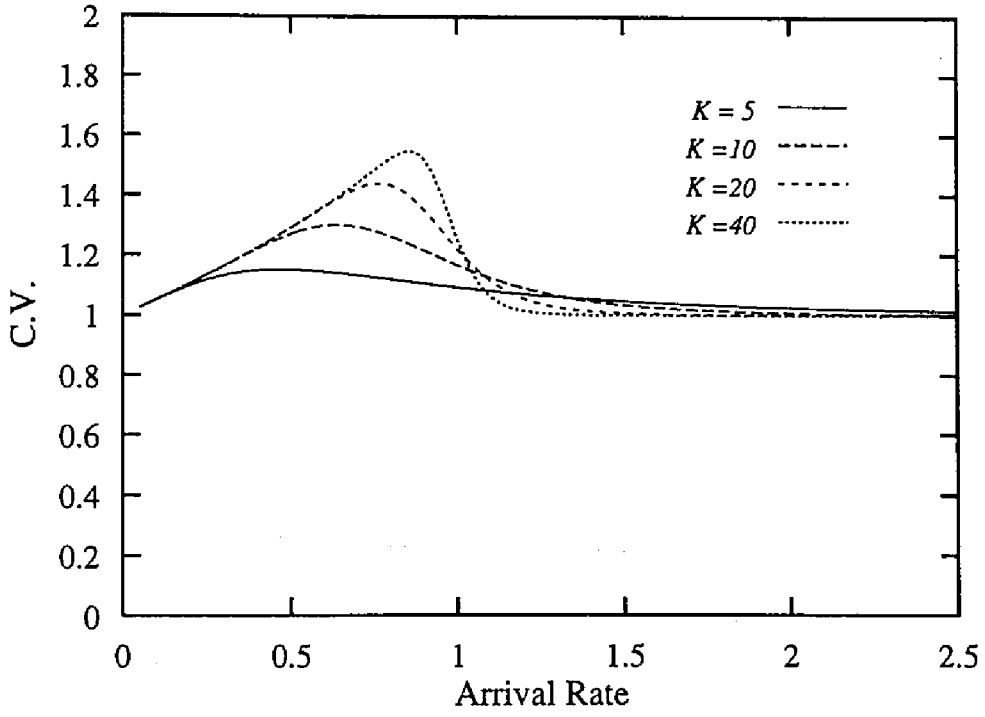


Figure 3.11: C.V. under Random Scheduling ($k = 1$, $v = 1$)

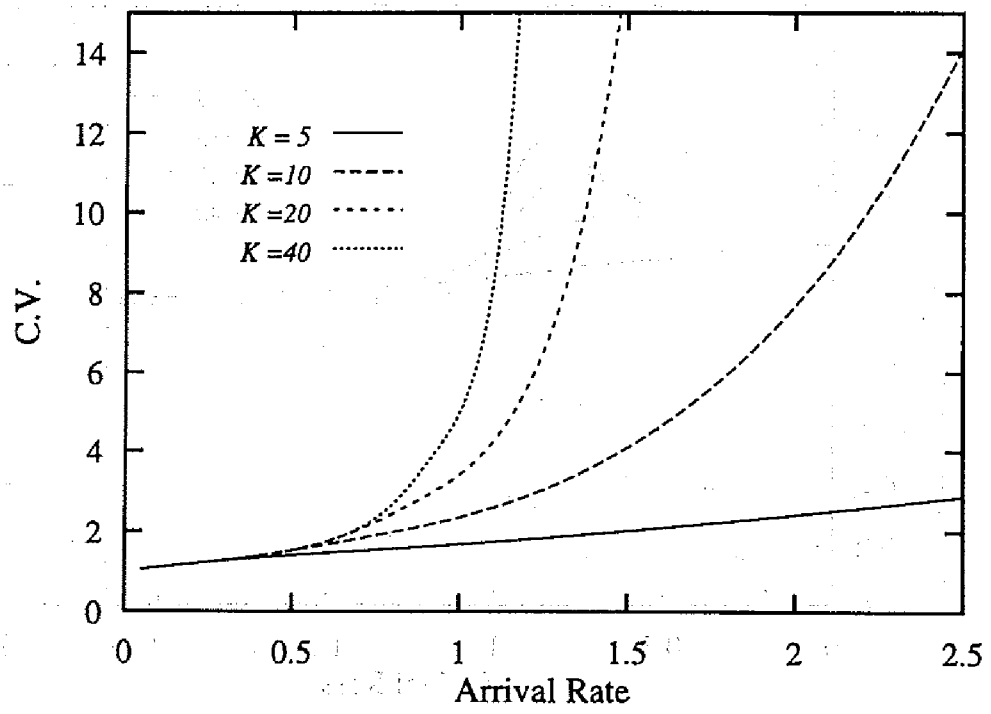


Figure 3.12: C.V. under LCFS ($k = 1$, $v = 1$)

Chapter 4

M/G/1/K with Push-out Scheme under Vacation Policy

4.1 Introduction

This chapter considers a queueing system with a finite buffer and server vacation. Messages are admitted into the system in accordance with an appropriate buffering policy. That is, a finite number of messages can be held in the system at any time since the system has a buffer of a finite capacity. There are two control policies for processing messages. One is the buffering policy by which messages are selected for admission into the system. The other is the service policy by which messages are selected for admission into the service facility.

Buffering policies specify those messages that are admitted to enter and those to be removed from the buffer instead when the buffer is full. Rubin and Ouaily [Rubi88] classified the buffering policies into the following types(Fig.4.1).

- Non-Preemptive-Buffering (NPB)
  An arriving message that finds the system full is blocked.

- Preemptive-Buffering (PB)
  If an arriving message finds the system full, the message which has waited the longest is pushed out from the buffer and the arriving message is allocated a buffer space.

The service policy determines the selection of messages waiting for service when the service facility becomes available. This policy includes, for example, FCFS, LCFS and random order of service.

Queueing systems with a finite buffer and server vacation have been extensively studied to model and analyze a number of computer communication systems. In particular, queueing systems with buffering policy have many applications like time-critical message transmission, sensor telemetry, radar communication and processing systems. In those applications, the information content of a message is associated with a timeliness index, so that the most recent message to arrive contains the most valuable information, and thus needs to be given preference for selection for service. On the other hand, the data transmission is the primary job for those systems and when there are no messages in the buffer, they start secondary jobs like testing and maintenance work. From a queueing theoretical point of view, those periods spent for the service of secondary jobs are considered as vacations.

Recently, with the increase of demands for multi-media communication, many protocols and architectures to accommodate traffics of different characteristics from multiple sources in

a common channel have been proposed and implemented so far [Arnb87, Turn86, Part94]. In this communication environment, messages are classified from two orthogonal points of view, delay and loss probability [Sumi88]. Delay (Loss probability) sensitive messages are insensitive to loss probability (delay) in general. These two factors can be expressed by assigning timeliness index to each message, which means after some critical value for its delay, each message becomes useless. For effective transmission of two types of messages, switching systems require the use of finite preemptive buffering service system since it is essential to provide short waiting time to those messages which are delay sensitive. If we focus our attention on the behavior of delay sensitive messages, the transmission of loss sensitive messages are considered as a secondary job for those switching systems. Thus, we can apply our model to evaluate the behavior of delay sensitive messages.

There are several literatures concerning buffering policies. A communication system under a preemptive buffering was investigated by Rubin and Ouaily in the context of an $M/G/1/K$ with push-out scheme [Rubi88]. Kröner analyzed loss probabilities for a partial buffer sharing scheme under FCFS [Krön90]. Sumita and Ozawa analyzed loss probabilities and the waiting time of systems with a push-out scheme [Sumi88].

Concerning queueing systems with server vacation, there are a number of previous works. An excellent survey of queueing systems with vacations, including some applications, was written by Doshi [Dosh86, Dosh90]. An $M/G/1/K$ with multiple vacation has also been analyzed by Lee [Lee84], but no analytical results are available for the model with push-out scheme.

This chapter is organized as follows: In section 4.2, we describe our mathematical model in detail. In section 4.3, we derive the relation of the mean waiting times for NPB, PB-served and PB-pushed-out messages. We also summarize Lee's results [Lee84] to obtain the joint probability distributions for the number of messages in the system and the remaining service or vacation time. In section 4.4, the LST of the waiting time distribution for an eventually served message is derived. In section 4.5, we show the numerical results.

4.2 Model

We consider an $M/G/1/K$ push-out model with multiple vacations (Fig.4.2). Messages arrive at the system according to a Poisson process with rate λ. The service time distribution function and its LST are denoted by $S(x)$ and $S^*(s)$, respectively. The mean service time is $1/\mu$.

When the system becomes idle, the server takes a vacation. The vacation policy of our model is multiple vacations. The server takes vacations repeatedly until he finds at least one waiting message accommodated upon returning from a vacation. The vacation time distribution function and its LST are denoted by $V(x)$ and $V^*(s)$, respectively. The mean vacation time is $1/v$.

The maximum number of messages that can be present in the system is $K < \infty$. When a message is in service, the maximum number of messages in the buffer is $K-1$. The buffering policy determines which to discard out of $K-1$ messages (K messages) to accommodate a newly arriving message when the server is busy (taking a vacation) and the system is full.

The buffering policy considered here is that when a new message finds the system full, a message with the longest sojourn time in the buffer is pushed out and lost.

We deal with two service disciplines, FCFS and LCFS.

4.3 Mean Waiting Time

Following the approach of [Rubi88], we consider the relation between the mean waiting time of NPB model and that of PB model. Let W_P denote the waiting time during which a message stays in the buffer in PB model. We then have

$$E[W_P] = E[W_P|\text{served}]Prob[\text{served}] + E[W_P|\text{pushed-out}]Prob[\text{pushed-out}]. \quad (4.1)$$

Let γ denote the system throughput. In both NPB and PB models, the event that a message is lost occurs when the system is full. Note that the stochastic behavior of the number of messages in the system does not depend on our buffering policy. Hence, L, γ and ρ' are invariant in the NPB and PB models with multiple vacations. Let W_B be the waiting time of a message accepted in the NPB model. Applying Little's theorem to those messages present in the queue, we have

$$\gamma E[W_B] = E[L] - \rho' = \lambda E[W_P]. \quad (4.2)$$

Since $\gamma \leq \lambda$, it follows that

$$E[W_P] \leq E[W_B]. \quad (4.3)$$

Considering the throughput γ, we have

$$\gamma = \lambda(1 - P_B) = \lambda(1 - Prob[\text{pushed-out}]). \quad (4.4)$$

Hence, we obtain

$$Prob[\text{pushed-out}] = P_B, \quad (4.5)$$

and

$$Prob[\text{served}] = 1 - Prob[\text{pushed-out}] = 1 - P_B. \quad (4.6)$$

Substituting (4.5) and (4.6) into (4.1), we have

$$\lambda E[W_P] = \lambda(1 - P_B)E[W_P|\text{served}] + \lambda P_B E[W_P|\text{pushed-out}]. \quad (4.7)$$

From (4.2) and (4.4), we obtain

$$\lambda E[W_P] = \lambda(1 - P_B)E[W_B]. \quad (4.8)$$

From (4.7) and (4.8), $E[W_P|\text{pushed-out}]$ is given by

$$E[W_P|\text{pushed-out}] = \frac{1 - P_B}{P_B}(E[W_B] - E[W_P|\text{served}]). \quad (4.9)$$

Thus, we can calculate the mean sojourn time of a pushed out message from (4.9) if we obtain $E[W_P|\text{served}]$.

4.4 Waiting Time Distribution for Served Messages

4.4.1 FCFS

We first consider the push-out system under FCFS service discipline. Each arriving message joins the queue at the tail and if the system is full upon arrival, the message at the head of the queue is pushed out.

Let $W_{k:n}$ denote the waiting time of a tagged message that has k other messages ahead and n others behind it at the end of a service or a vacation. We also define the following LST:

$$W^*_{k:n}(s) = E[e^{-sW_{k:n}}|\text{served}]Prob[\text{served}], \tag{4.10}$$

where $0 \le k \le K-1$ and $0 \le n \le K-k-1$ at the end of a vacation, and $0 \le k \le K-2$ and $0 \le n \le K-k-2$ at the end of a service. Note that the LST $W^*_{k:n}(s)$ is the same in the both cases of a vacation and a service.

The set $\{W^*_{k:n}(s); 0 \le k \le K-1,\ 0 \le n \le K-k-1\}$ satisfies the following equations:

$$W^*_{0:n}(s) = 1, \qquad 0 \le n \le K-2, \tag{4.11}$$

$$W^*_{k:n}(s) = \sum_{j=0}^{K-k-n-1} S^*_j(s) \cdot W^*_{k-1:n+j}(s) + \sum_{j=K-k-n}^{K-n-2} S^*_j(s) \cdot W^*_{K-n-j-2:n+j}(s),$$
$$1 \le k \le K-1,\ 0 \le n \le K-k-1, \tag{4.12}$$

where $S^*_j(s)$ is defined in (2.34). Using these LSTs, the LST $W^*(s)$ of the distribution function for the waiting time of a served message in the FCFS system is given by

$$\begin{aligned} W^*(s) = \frac{1}{1-P_B} \Bigg[& \sum_{j=0}^{K-1} \Bigg\{ \sum_{k=0}^{K-j-2} \Omega^*_{j:k}(s) \cdot W^*_{j:k}(s) \\ & + \sum_{k=K-j-1}^{K-1} \Omega^*_{j:k}(s) \cdot W^*_{K-k-1:k}(s) \Bigg\} + \sum_{k=0}^{K-1} \Omega^*_{K:k}(s) \cdot W^*_{K-k-1:k}(s) \\ & + \sum_{j=1}^{K-1} \Bigg\{ \sum_{k=0}^{K-j-1} \Pi^*_{j:k}(s) \cdot W^*_{j-1:k}(s) + \sum_{k=K-j}^{K-2} \Pi^*_{j:k}(s) \cdot W^*_{K-k-2:k}(s) \Bigg\} \\ & + \sum_{k=0}^{K-2} \Pi^*_{K:k}(s) \cdot W^*_{K-k-2:k}(s) \Bigg], \end{aligned} \tag{4.13}$$

where $\Omega^*_{j:k}(s)$ and $\Pi^*_{j:k}(s)$ are defined in (3.19), (3.20), (3.23) and (3.23) of chapter 3.

In [Rubi88], there is a technical error. The waiting time distribution of a served message $W(t)$ is given by

$$W(t) = \pi_0 + \sum_{n=1}^{K} \pi_n [R(t) * B^{(n-1)}(t)],$$

where π_n's are the steady state probabilities that an arriving message finds n messages in the system, $B(t)$ is the service time distribution, $R(t)$ is the remaining service time distribution, $*$ denotes the convolution and $B^{(n-1)}(t)$ is the $n-1$st convolution. In that equation, the number of messages at an arriving epoch and the remaining service time are treated as being independent, but that is wrong. The number of messages at an arriving epoch is not independent of the remaining service time. Thus, we have to use the joint distribution of the number of messages and the remaining service time. (We show the corrected LST of the waiting time distribution in Appendix D.)

4.4.2 LCFS

We next consider the LCFS system. Each arriving message joins the queue at the head and if the system is full, the message at the tail is pushed out.

As in the case of FCFS, let $\widetilde{W}_k$ denote the waiting time of a tagged message that has k other messages ahead at the end of a service or a vacation. We define the following LST:

$$\widetilde{W}_k^*(s) = E[e^{-s\widetilde{W}_k}|\text{served}]Prob[\text{served}], \tag{4.14}$$

where $0 \leq k \leq K-1$ at the end of a vacation, and $0 \leq k \leq K-2$ at the end of a service. Note that $\widetilde{W}_k^*(s)$ is the same in the cases of both a vacation and a service.

The set $\{\widetilde{W}_k^*(s);\ 0 \leq k \leq K-1\}$ satisfies the following equations:

$$\widetilde{W}_0^*(s) = 1, \tag{4.15}$$

$$\widetilde{W}_k^*(s) = \sum_{j=0}^{K-k-1} S_j^*(s) \cdot \widetilde{W}_{k+j-1}^*(s), \qquad 1 \leq k \leq K-1. \tag{4.16}$$

For simplicity, we define the following LSTs:

$$\widetilde{S}_j^*(s) = \int_0^\infty \frac{(\lambda x)^j}{j!} e^{-(s+\lambda)x} d\widetilde{S}(x), \tag{4.17}$$

$$\widetilde{V}_j^*(s) = \int_0^\infty \frac{(\lambda x)^j}{j!} e^{-(s+\lambda)x} d\widetilde{V}(x), \tag{4.18}$$

where $\widetilde{S}(x) = Prob[\widetilde{S} \leq x]$ and $\widetilde{V}(x) = Prob[\widetilde{V} \leq x]$. If k messages arrive at the system during the remaining vacation or service time, the tagged message has k messages ahead at the end of this vacation or service. Thus, we have the LST of the distribution function for the waiting time of a served message in the LCFS, $W^*(s)$ by

$$W^*(s) = \frac{1}{1-P_B}\left[(1-\rho')\sum_{k=0}^{K-1} \widetilde{V}_k^*(s) \cdot \widetilde{W}_k^*(s) + \rho' \sum_{k=0}^{K-2} \widetilde{S}_k^*(s) \cdot \widetilde{W}_k^*(s)\right]. \tag{4.19}$$

4.5 Numerical Results

In this section, we show the numerical results for the mean and the c.v. of the waiting time using the analysis presented in section 4.4.

In our numerical examples, we choose the system size K equal to 5, that is, the buffer size equals 4. As for vacation times, we assume an exponential distribution with mean 1.0. The mean service time is fixed at 1.0, and the performance values are calculated by changing the arrival rate.

First, we compare the mean waiting time under various situations. Using (4.9), (4.13) and (4.19), we calculate the mean waiting times for served and pushed-out messages. From [Lee84], the mean waiting time for NPB model can be also calculated.

Figs.4.3. to 4.4. illustrate the mean waiting time for three types of messages: NPB, PB-served and PB-pushed-out. Furthermore, mean waiting times under the exponential service distribution are compared with those under deterministic one.

In both figures, the mean waiting times of NPB and PB-served messages tend to the value of 1 as the offered load gets small. This is because each arriving message most likely waits for the remaining vacation time. On the other hand, the mean sojourn time of a pushed-out message is larger than those of others. This phenomenon can be explained as follows. When the arrival rate is very small, there are few messages in the system. Thus, most of arriving messages are eventually served. However, if an arriving message is eventually pushed out, its sojourn time becomes large due to light traffic.

Next, when the offered load gets large, the mean waiting times for all types of messages converge under exponential and constant service times, in particular, PB-served and PB-pushed-out messages converge to the same value. In both NPB and PB cases, a new arriving message which can be accommodated in the system finds four other messages (including the message in service) ahead when the arrival rate is very large. Hence, the mean waiting time of NPB messages converges to the value 4. In PB case, a new arriving message can enter the system. But there are many other new arriving messages behind that and the probability that the tagged message is eventually served gets small. Thus, the mean waiting times in the buffer of both messages become small.

In FCFS(Fig.4.3), the mean waiting time of a PB-pushed-out message is bounded by 5, because each arriving message finds at most five messages ahead. On the other hand, in LCFS(Fig.4.4), it may exceed 5. This is because there is no bound on the number of the messages which are served before the service of the tagged one.

In Fig.4.4, the mean waiting time of a PB-pushed-out message under the deterministic service time distribution fluctuates remarkably when the arrival rate is small. It can be considered that under deterministic service distribution, the mean waiting time of a PB-pushed-out message is influenced by the loss probability and the waiting time of a PB-served one.

One more interesting observation is the relation of the mean waiting time between PB-served and PB-pushed-out messages under two service time distributions. Let $W_{A:B}[C]$ denote the mean waiting time of a 'C' type message under 'A' service discipline and 'B' service distribution.

In Fig.4.3, it is observed that $W_{\text{FCFS:Exp}}[\text{Served}] \leq W_{\text{FCFS:Exp}}[\text{Pushed-out}]$, i.e., the mean waiting time of a served message is always smaller than that of a pushed-out one. On the other hand, under the deterministic service time distribution, we see that

$$W_{\text{FCFS:Det}}[\text{Served}] \leq W_{\text{FCFS:Det}}[\text{Pushed-out}], \qquad 0 \leq \rho \leq 1, \tag{4.20}$$

$$W_{\text{FCFS:Det}}[\text{Served}] > W_{\text{FCFS:Det}}[\text{Pushed-out}], \qquad \rho > 1. \tag{4.21}$$

In the LCFS case, we can observe the following relations:

$$W_{\text{LCFS:Exp}}[\text{Served}] < W_{\text{LCFS:Exp}}[\text{Pushed-out}], \qquad 0 \leq \rho, \tag{4.22}$$

$$W_{\text{LCFS:Det}}[\text{Served}] < W_{\text{LCFS:Det}}[\text{Pushed-out}], \qquad 0 \leq \rho. \tag{4.23}$$

Equations (4.22) and (4.23) show that the mean waiting time of the served message is always smaller than that of the pushed-out one under both service distributions. Thus, in FCFS, the mean waiting times of the served and pushed-out messages are more influenced by the type of service distribution.

In Fig.4.5 and Fig.4.6, the mean waiting times are compared for two push-out models; the system with vacation and that without vacation. We can calculate the mean waiting time of the system without vacation by [Rubi88](see Appendix D). In both figures, we assume $S(x)$ to be exponential (mean service time = 1.0). From both figures, we can observe the influence of vacations when the offered load is small. Furthermore, when the offered load becomes large, each mean waiting time converges to the same value. This is because taking vacations hardly affects the performance measures when the offered load is large.

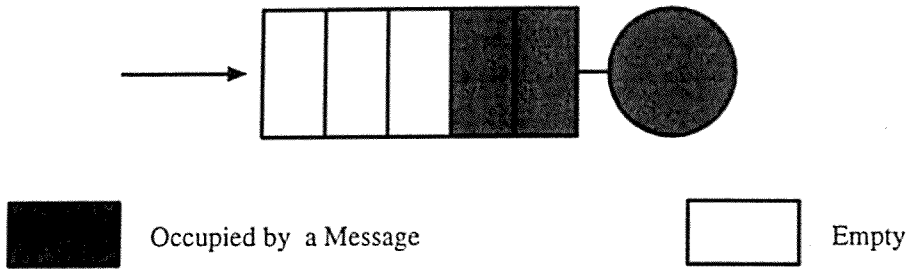
Fig.4.7 illustrates the c.v. of the waiting time of the PB-served message under two service time disciplines and two service distributions. In both FCFS and LCFS cases, the values start from 1 because the vacation distribution is exponential and its mean equals 1.0. We also observe that both curves converge rapidly. This means that the fluctuation of the waiting time is small when the offered load becomes large. We note that the variation under LCFS is larger than that under FCFS.

Fig.4.8 illustrates the c.v. of the waiting time for NPB and PB-served messages with and without vacations. When the offered load is small, the influence of vacations is recognized. In FCFS cases, all values converge to the same value when the offered load is large. On the other hand, in LCFS cases, the values of the PB-served message with and without vacations converge to the same value but that of NPB model diverges to infinity. We observe that the waiting time of the PB-served message with vacations varies least in both FCFS and LCFS cases.

4.6 Conclusion

In this chapter, we have considered a buffer controlling policy, called push-out scheme. We investigated the behaviors of the two types of messages, one is eventually served and the other is pushed out from the system.

From the numerical results, the following has been found. First, the mean waiting times of NPB and PB-served messages significantly depends on the remaining vacation time. In such a situation, the waiting time of the PB-pushed-out message is larger than others. The mean waiting times of PB-served and PB-pushed out messages converge as the arrival rate gets large, and those limiting values are smaller than that under NPB case. This is due to the push-out scheme. We found that the mean waiting times under PB case are influenced by the service time distribution. Furthermore, the variation of the waiting time of the PB-served message is small and stable in comparison with that of the NPB one.



(a) NPB Model

(b) PB Model

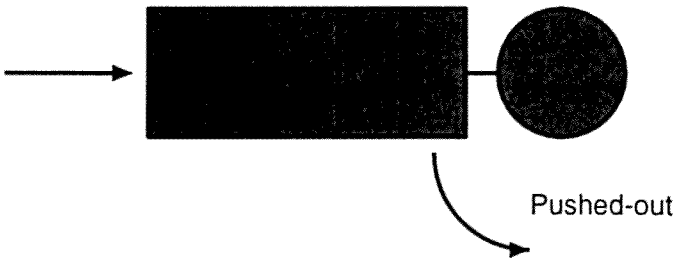


Figure 4.1: NPB and PB Models

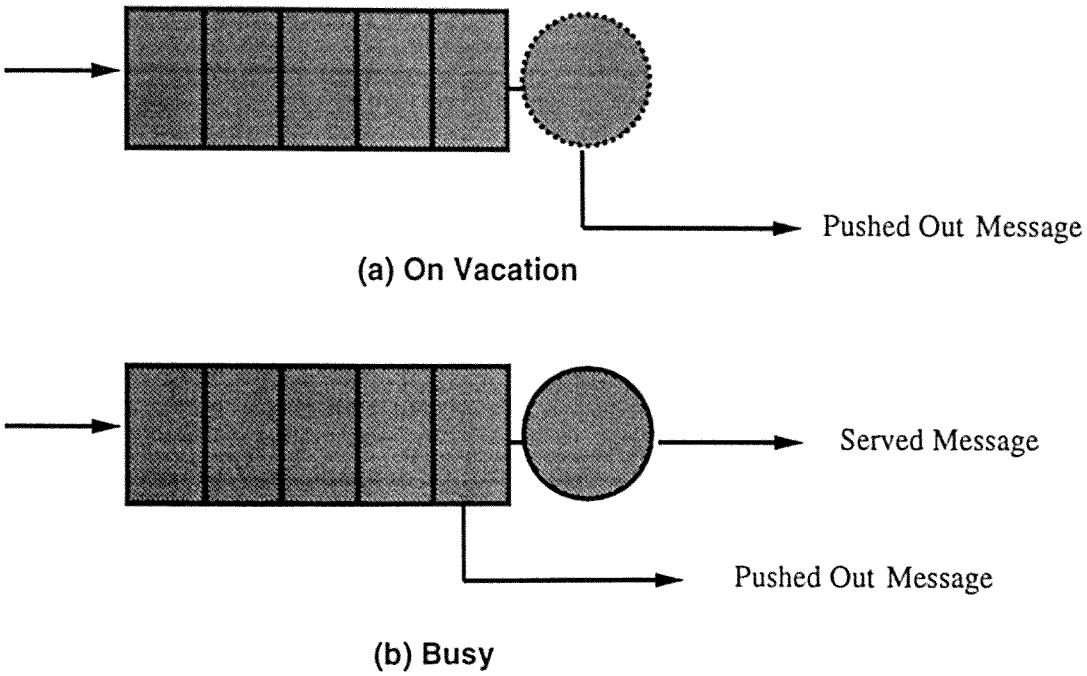


Figure 4.2: Push-Out Model with Vacation

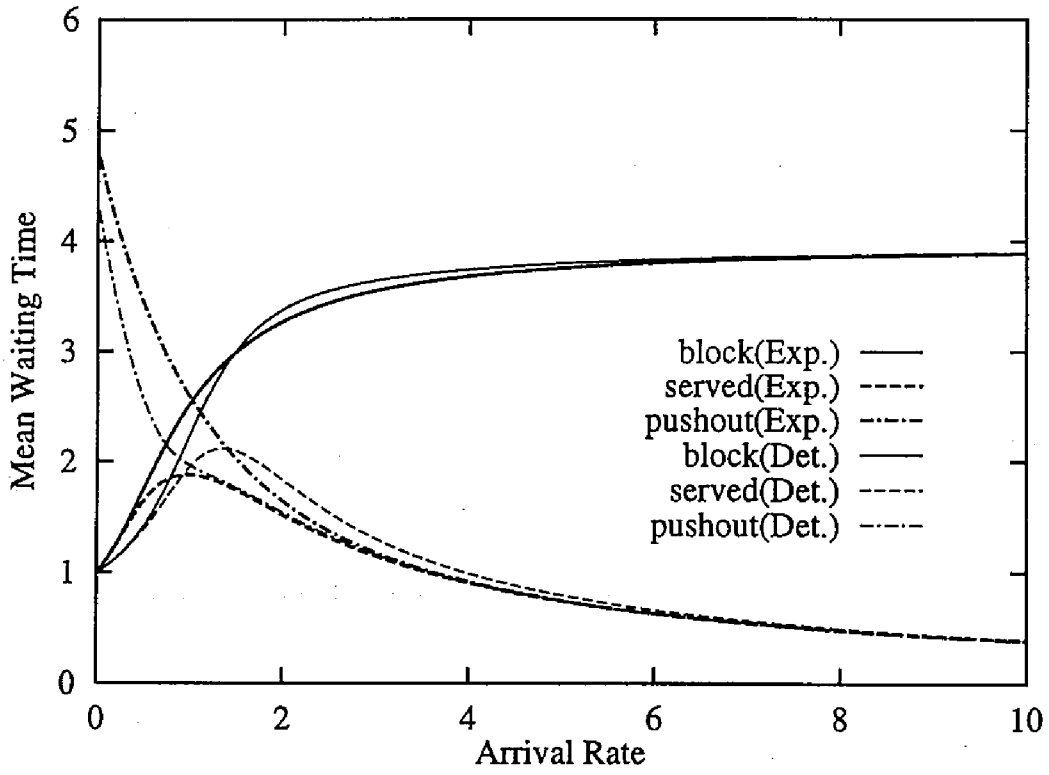


Figure 4.3: Mean Waiting Time under FCFS

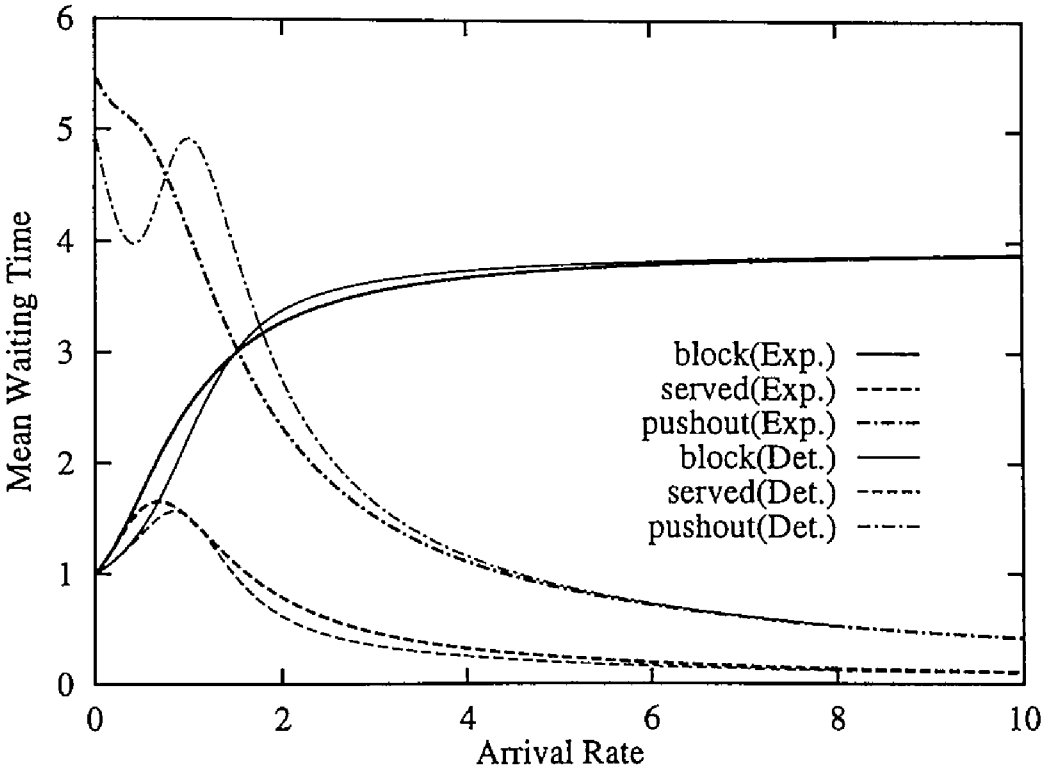


Figure 4.4: Mean Waiting Time under LCFS

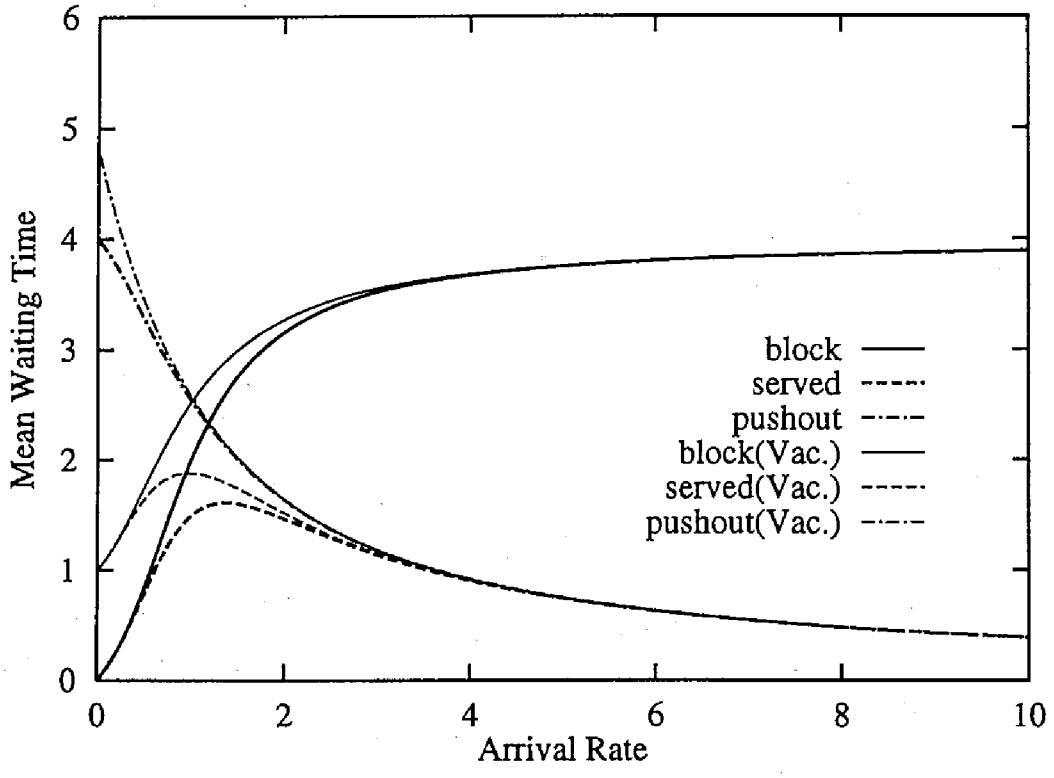


Figure 4.5: Mean Waiting Time under FCFS (non-Vac. v.s. Vac.)

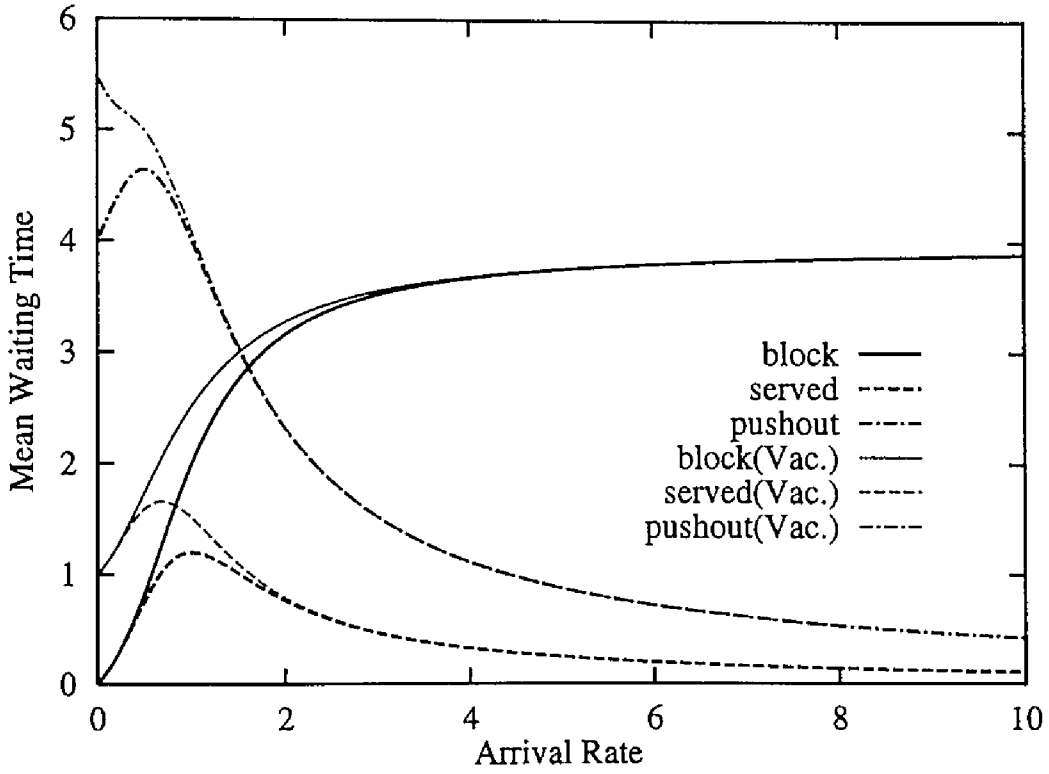


Figure 4.6: Mean Waiting Time under LCFS (non-Vac. v.s. Vac.)

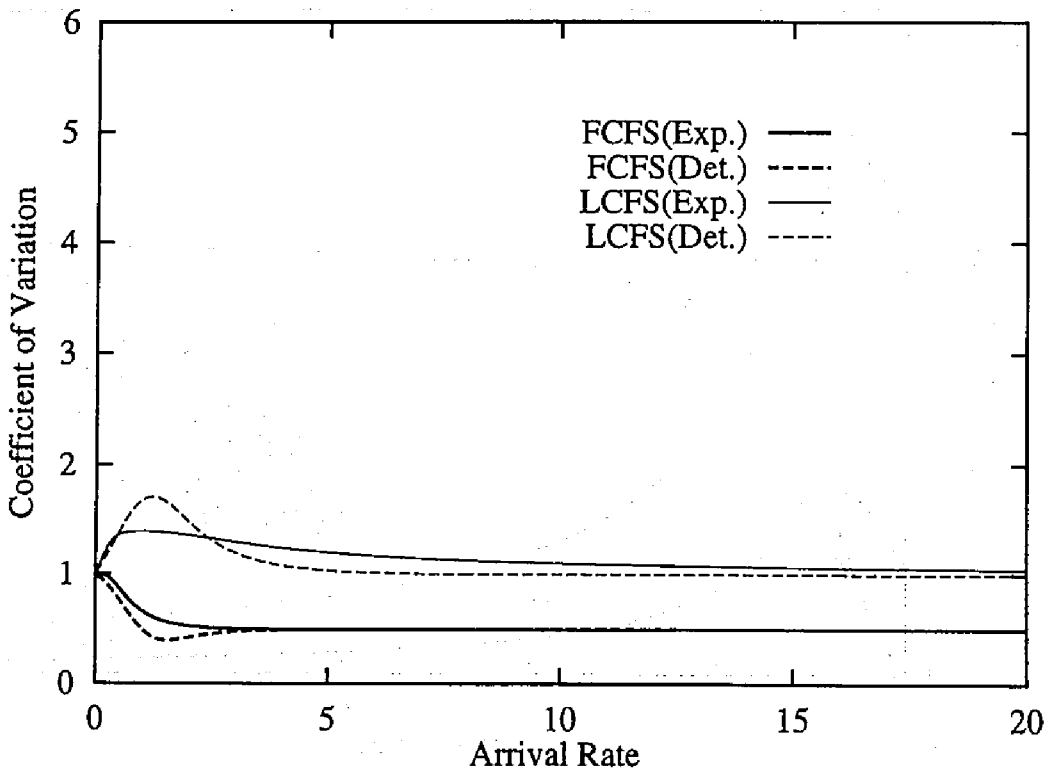


Figure 4.7: C.V. of Waiting Time

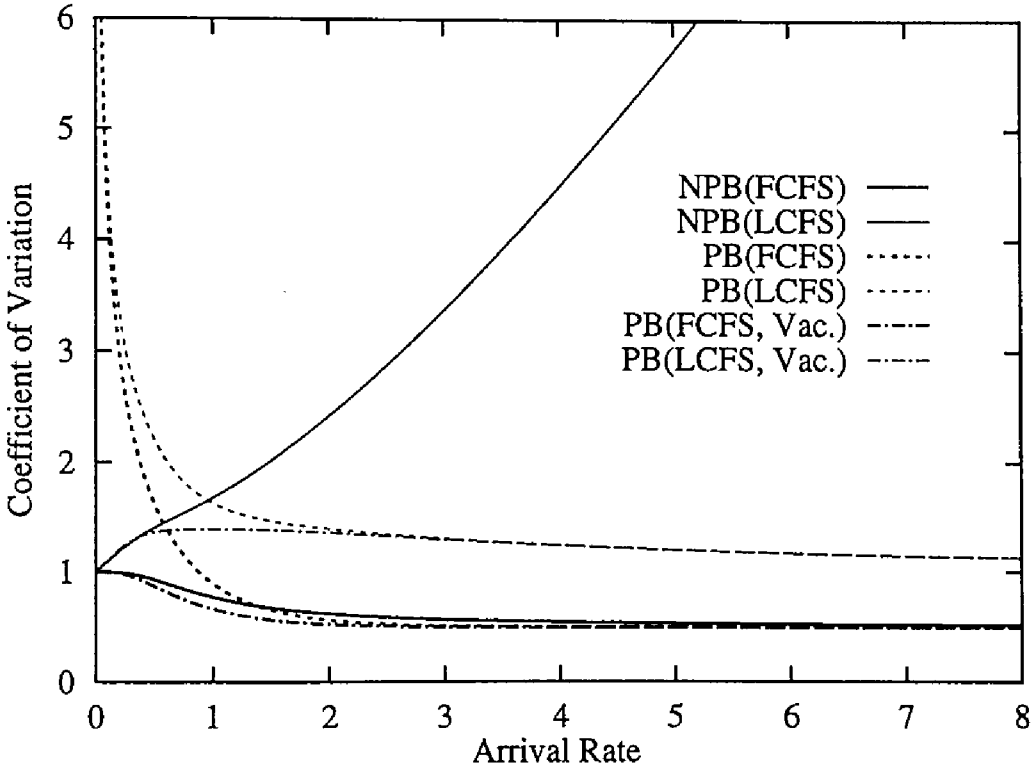


Figure 4.8: C.V. of Waiting Time

Chapter 5

SPP/G/1 with Multiple Vacations and E-limited Service Discipline

5.1 Introduction

Most of studies of vacation models have been related with $M/G/1$ systems. That is, messages arrive to the system in accordance with a Poisson process, service times are independent and identically distributed (i.i.d.) according to a general probability distribution function. Those studies have explicitly analyzed some of the performance measures, such as queue length, waiting time, and so on.

As Asynchronous Transfer Mode (ATM) becomes important as one of the key technologies for broadband ISDN, many papers related to the performance analysis of ATM switching fabrics have appeared [Blon91, Sumi88]. Since the traffic stream has the property of burstiness, the arrival process cannot be modeled as a Poisson process.

Recently, queueing systems with vacations and non-Poissonian arrivals have been studied. Lucantoni et al. [Luca90] have analyzed a single-server queue with multiple vacations, where the input process is the *MAP*. The *MAP* is a particularly tractable point process and includes the *MMPP* and the phase-type renewal process. Neuts [Neut81, Neut89] developed the matrix analytical approach for the *MAP*. Blondia [Blon91] has considered a single server queue with a finite waiting room where the server takes vacations and analyzed the model for both the cases under the exhaustive and the limited service disciplines. Concerning the model, no explicit formulas and numerical results for the performance measures like the mean waiting time have been presented.

In this chapter, we consider a queueing system with multiple vacations and E-limited service discipline where the message arrival process is an *SPP*. The *SPP* is a two-state *MMPP* and hence performance measures like the queue length and the mean waiting time can be derived explicitly.

The arrival process of messages is an *SPP* which is modulated by a continuous-time Markov chain with two states 1 and 2. Time spent in state 1 (2) is exponentially distributed with rate α (β). Let ζ denote the state in the underlying Markov process. When $\zeta = i$ $(i = 1, 2)$, messages arrive to the system according to a Poisson process with parameter λ_i. Thus the mean arrival rate λ is given by

$$\lambda = \frac{\beta\lambda_1 + \alpha\lambda_2}{\alpha + \beta}.$$

Message service times are i.i.d. according to a general probability distribution $S(x)$ whose LST

is denoted by $S^*(s)$.

The server serves messages under the E-limited service discipline. Before taking a vacation, the server continues to serve until at most M messages are served or the system becomes empty, whichever occurs first. On return from a vacation, if the server finds the system empty, he takes another vacation. If the server finds at least one message in the system, he begins serving the waiting messages. The system is called a multiple vacation model. Vacation times are i.i.d. according to a general probability distribution. Let $V(x)$ and $V^*(s)$ be the PDF and its LST of a vacation time V, respectively.

Let S denote a random variable for a message service time. All messages arriving to the system are eventually served. That is, the system has a buffer of an infinite capacity and the following inequality is satisfied (see Appendix E.2):

$$\rho + \lambda E[V]/M \ < \ 1,$$

where $\rho = \lambda E[S]$. Service is nonpreemptive: once selected for service, a message is served to completion continuously. Further, the service order of messages is independent of their service times.

Throughout the chapter, we assume that the system is in equilibrium. For simplicity, we assume that $S(0) = 0$ and $V(0) = 0$, and that the PDF's $S(x)$ and $V(x)$ are absolutely continuous with the pdf's $h(x)$ and $v(x)$, respectively.

The remainder of this chapter is organized as follows. In section 5.2, we obtain the joint distribution of queue length and either of the elapsed service or vacation time. In section 5.3, we derive the mean queue length and the mean waiting time.

5.2 Queue Length Distribution

In this section, we consider the joint distribution of the queue length, the state of the server and of the arrival process, and the elapsed service time for a message if the server is busy, or the elapsed vacation time if the server is on vacation.

First, we define the following notations:

$$\xi \ = \ \begin{cases} 0 & \text{if the server is on vacation.} \\ m & \text{if the server is busy and serving the } m\text{-th message after taking the last vacation, } (1 \le m \le M). \end{cases}$$

ζ = state of the arrival process.

L = number of messages present in the system.

$\widehat{S}$ = elapsed service time for a message in service.

$\widehat{V}$ = elapsed vacation time for the server on vacation.

The joint pdf's $P_{k,m}^{(l)}(x)$ and $Q_k^{(l)}(x)$ are defined as

$$P_{k,m}^{(l)}(x)dx \ = \ Prob\{L = k, \xi = m, \zeta = l, x < \widehat{S} < x + dx\}, \quad (x \ge 0,\ k \ge 1,\ l = 1,2), \tag{5.1}$$

$$Q_k^{(l)}(x)dx \ = \ Prob\{L = k, \xi = 0, \zeta = l, x < \widehat{V} < x + dx\}, \quad (x \ge 0,\ k \ge 0,\ l = 1,2). \tag{5.2}$$

These pdf's satisfy the following equations:

$$\frac{d}{dx}P_{k,m}^{(1)}(x) \ = \ -(\lambda_1 + \frac{h(x)}{1 - S(x)} + \alpha)P_{k,m}^{(1)}(x) + \lambda_1 P_{k-1,m}^{(1)}(x) + \beta P_{k,m}^{(2)}(x), \tag{5.3}$$

$$\frac{d}{dx}P_{k,m}^{(2)}(x) = -(\lambda_2 + \frac{h(x)}{1-S(x)} + \beta)P_{k,m}^{(2)}(x) + \lambda_2 P_{k-1,m}^{(2)}(x) + \alpha P_{k,m}^{(1)}(x), \tag{5.4}$$

$$(x \geq 0,\ k \geq 1,\ 1 \leq m \leq M),$$

$$P_{k,m}^{(l)}(0) = \int_0^\infty P_{k+1,m-1}^{(l)}(x) \cdot \frac{h(x)}{1-S(x)}dx, \qquad (k \geq 1,\ 2 \leq m \leq M), \tag{5.5}$$

$$P_{k,1}^{(l)}(0) = \int_0^\infty Q_k^{(l)}(x) \cdot \frac{v(x)}{1-V(x)}dx, \qquad (k \geq 1), \tag{5.6}$$

$$\frac{d}{dx}Q_k^{(1)}(x) = -(\lambda_1 + \frac{v(x)}{1-V(x)} + \alpha)Q_k^{(1)}(x) + \lambda_1 Q_{k-1}^{(1)}(x) + \beta Q_k^{(2)}(x), \tag{5.7}$$

$$\frac{d}{dx}Q_k^{(2)}(x) = -(\lambda_2 + \frac{v(x)}{1-V(x)} + \beta)Q_k^{(2)}(x) + \lambda_2 Q_{k-1}^{(2)}(x) + \alpha Q_k^{(1)}(x), \tag{5.8}$$

$$(x \geq 0,\ k \geq 0),$$

$$Q_0^{(l)}(0) = \int_0^\infty Q_0^{(l)}(x) \cdot \frac{v(x)}{1-V(x)}dx + \sum_{m=1}^{M}\int_0^\infty P_{1,m}^{(l)}(x) \cdot \frac{h(x)}{1-S(x)}dx, \tag{5.9}$$

$$Q_k^{(l)}(0) = \int_0^\infty P_{k+1,M}^{(l)}(x) \cdot \frac{h(x)}{1-S(x)}dx, \qquad (k \geq 1), \tag{5.10}$$

and

$$\sum_{k=1}^{\infty}\sum_{m=1}^{M}\sum_{l=1}^{2}\int_0^\infty P_{k,m}^{(l)}(x)dx + \sum_{k=0}^{\infty}\sum_{l=1}^{2}\int_0^\infty Q_k^{(l)}(x)dx = 1, \tag{5.11}$$

where we assume $P_{0,m}^{(l)}(x) \equiv 0$ and $Q_{-1}^{(l)}(x) \equiv 0$ for $l = 1, 2$.

For derivation of $Q_k^{(l)}(x)$, we define $\overline{Q}_k^{(l)}(x)$ as

$$\overline{Q}_k^{(l)}(x) = \frac{Q_k^{(l)}(x)}{1-V(x)}, \qquad (k \geq 0,\ x \geq 0,\ l = 1, 2). \tag{5.12}$$

Then, from (5.7) and (5.8), we obtain

$$\frac{d}{dx}\overline{Q}_k^{(1)}(x) = -(\lambda_1 + \alpha)\overline{Q}_k^{(1)}(x) + \lambda_1 \overline{Q}_{k-1}^{(1)}(x) + \beta\overline{Q}_k^{(2)}(x), \tag{5.13}$$

$$\frac{d}{dx}\overline{Q}_k^{(2)}(x) = -(\lambda_2 + \beta)\overline{Q}_k^{(2)}(x) + \lambda_2 \overline{Q}_{k-1}^{(2)}(x) + \alpha\overline{Q}_k^{(1)}(x), \tag{5.14}$$

$$(k \geq 0,\ x \geq 0).$$

Multiplying both sides in (5.13) and (5.14) by z^k ($|z| \leq 1$) and summing over all $k \geq 0$ yields

$$\frac{\partial}{\partial x}\begin{pmatrix} \overline{Q}^{(1)}(z,x) \\ \overline{Q}^{(2)}(z,x) \end{pmatrix} = \begin{pmatrix} -\lambda_1(z) - \alpha & \beta \\ \alpha & -\lambda_2(z) - \beta \end{pmatrix}\begin{pmatrix} \overline{Q}^{(1)}(z,x) \\ \overline{Q}^{(2)}(z,x) \end{pmatrix}, \tag{5.15}$$

where, for $l = 1, 2$,

$$\overline{Q}^{(l)}(z,x) = \sum_{k=0}^{\infty}\overline{Q}_k^{(l)}(x)z^k, \tag{5.16}$$

and

$$\lambda_l(z) = \lambda_l - \lambda_l z. \tag{5.17}$$

The general solution of the partial differential equations (5.15) is found to be

$$\begin{pmatrix} \overline{Q}^{(1)}(z,x) \\ \overline{Q}^{(2)}(z,x) \end{pmatrix} = \begin{pmatrix} e^{-p(z)x} & \hat{q}(z)e^{-q(z)x} \\ \hat{p}(z)e^{-p(z)x} & e^{-q(z)x} \end{pmatrix}\begin{pmatrix} K_1(z) \\ K_2(z) \end{pmatrix}, \tag{5.18}$$

where $K_1(z)$ and $K_2(z)$ are functions of z and

$$p(z), q(z) = \frac{\lambda_1(z) + \lambda_2(z) + \alpha + \beta \mp \sqrt{(\lambda_1(z) + \alpha - \lambda_2(z) - \beta)^2 + 4\alpha\beta}}{2}, \tag{5.19}$$

and

$$\hat{p}(z) = \frac{\lambda_1(z) + \alpha - p(z)}{\beta}, \tag{5.20}$$

$$\hat{q}(z) = \frac{\lambda_2(z) + \beta - q(z)}{\alpha}. \tag{5.21}$$

Now, we define the probability generating function of $Q_k^{(l)}(x)$'s as

$$Q^{(l)}(z, x) = \sum_{k=0}^{\infty} Q_k^{(l)}(x) \cdot z^k. \tag{5.22}$$

Note that

$$\overline{Q}^{(l)}(z, 0) = Q^{(l)}(z, 0). \tag{5.23}$$

Substituting $x = 0$ in (5.18), we obtain

$$\begin{pmatrix} \overline{Q}^{(1)}(z,0) \\ \overline{Q}^{(2)}(z,0) \end{pmatrix} = \begin{pmatrix} 1 & \hat{q}(z) \\ \hat{p}(z) & 1 \end{pmatrix} \begin{pmatrix} K_1(z) \\ K_2(z) \end{pmatrix}. \tag{5.24}$$

Thus $K_1(z)$ and $K_2(z)$ can be expressed in terms of $Q^{(1)}(z, 0)$ and $Q^{(2)}(z, 0)$ as

$$\begin{pmatrix} K_1(z) \\ K_2(z) \end{pmatrix} = \frac{1}{1 - \hat{p}(z)\hat{q}(z)} \begin{pmatrix} 1 & -\hat{q}(z) \\ -\hat{p}(z) & 1 \end{pmatrix} \begin{pmatrix} Q^{(1)}(z,0) \\ Q^{(2)}(z,0) \end{pmatrix}. \tag{5.25}$$

Then, (5.18) becomes

$$\begin{pmatrix} \overline{Q}^{(1)}(z,x) \\ \overline{Q}^{(2)}(z,x) \end{pmatrix} = \frac{1}{1 - \hat{p}(z)\hat{q}(z)} \begin{pmatrix} e^{-p(z)x} & \hat{q}(z)e^{-q(z)x} \\ \hat{p}(z)e^{-p(z)x} & e^{-q(z)x} \end{pmatrix} \cdot \begin{pmatrix} 1 & -\hat{q}(z) \\ -\hat{p}(z) & 1 \end{pmatrix} \begin{pmatrix} Q^{(1)}(z,0) \\ Q^{(2)}(z,0) \end{pmatrix}. \tag{5.26}$$

For derivation of $P_{k,m}^{(l)}(x)$ $(k \geq 0,\ 1 \leq m \leq M,\ l = 1, 2)$, we define the following generating functions:

$$P_m^{(l)}(z, x) = \sum_{k=1}^{\infty} P_{k,m}^{(l)}(x) \cdot z^k, \qquad (1 \leq m \leq M), \tag{5.27}$$

$$\overline{P}_m^{(l)}(z, x) = \frac{P_m^{(l)}(z, x)}{1 - S(x)}, \qquad (1 \leq m \leq M). \tag{5.28}$$

We note that

$$\overline{P}_m^{(l)}(z, 0) = P_m^{(l)}(z, 0), \qquad (l = 1, 2). \tag{5.29}$$

Similarly to $Q_k^{(l)}(x)$, $\overline{P}_m^{(1)}(z,x)$ and $\overline{P}_m^{(2)}(z,x)$ are found in terms of $P_m^{(1)}(z,0)$ and $P_m^{(2)}(z,0)$ as

$$\begin{pmatrix} \overline{P}_m^{(1)}(z,x) \\ \overline{P}_m^{(2)}(z,x) \end{pmatrix} = \frac{1}{1-\hat{p}(z)\hat{q}(z)} \begin{pmatrix} e^{-p(z)x} & \hat{q}(z)e^{-q(z)x} \\ \hat{p}(z)e^{-p(z)x} & e^{-q(z)x} \end{pmatrix} \cdot \begin{pmatrix} 1 & -\hat{q}(z) \\ -\hat{p}(z) & 1 \end{pmatrix} \begin{pmatrix} P_m^{(1)}(z,0) \\ P_m^{(2)}(z,0) \end{pmatrix}, \quad (1 \le m \le M). \tag{5.30}$$

Now we yield $Q^{(l)}(z,0)$ $(l=1,2)$ and $P_m^{(l)}(z,0)$ $(1 \le m \le M,\ l=1,2\ )$ explicitly. First, we consider the boundary conditions (5.5), (5.6), (5.9) and (5.10). Multiplying both sides of these equations by z^k $(|z| \le 1)$ and summing over all k, we obtain

$$P_1^{(l)}(z,0) = \int_0^\infty \overline{Q}^{(l)}(z,x)dV(x) - \int_0^\infty \overline{Q}^{(l)}(0,x)dV(x), \tag{5.31}$$

$$P_m^{(l)}(z,0) = \frac{1}{z}\int_0^\infty \overline{P}_{m-1}^{(l)}(z,x)dS(x) - \int_0^\infty P_{1,m-1}^{(l)}(x)\frac{h(x)}{1-S(x)}dx, \ (2 \le m \le M), \tag{5.32}$$

$$Q^{(l)}(z,0) = \int_0^\infty \overline{Q}^{(l)}(0,x)dV(x) + \frac{1}{z}\int_0^\infty \overline{P}_M^{(l)}(z,x)dS(x) + \sum_{m=1}^{M-1}\int_0^\infty P_{1,m}^{(l)}(x)\frac{h(x)}{1-S(x)}dx. \tag{5.33}$$

For the calculation of the above functions, we introduce the following matrices and vectors:

$$A(z) = \frac{1}{z(1-\hat{p}(z)\hat{q}(z))} \begin{pmatrix} S^*(p(z)) & \hat{q}(z)S^*(q(z)) \\ \hat{p}(z)S^*(p(z)) & S^*(q(z)) \end{pmatrix} \begin{pmatrix} 1 & -\hat{q}(z) \\ -\hat{p}(z) & 1 \end{pmatrix}, \tag{5.34}$$

$$B(z) = \frac{1}{1-\hat{p}(z)\hat{q}(z)} \begin{pmatrix} V^*(p(z)) & \hat{q}(z)V^*(q(z)) \\ \hat{p}(z)V^*(p(z)) & V^*(q(z)) \end{pmatrix} \begin{pmatrix} 1 & -\hat{q}(z) \\ -\hat{p}(z) & 1 \end{pmatrix}, \tag{5.35}$$

$$\boldsymbol{P}_m(z,x) = \begin{pmatrix} \overline{P}_m^{(1)}(z,x) \\ \overline{P}_m^{(2)}(z,x) \end{pmatrix}, \quad \boldsymbol{Q}(z,x) = \begin{pmatrix} \overline{Q}^{(1)}(z,x) \\ \overline{Q}^{(2)}(z,x) \end{pmatrix}, \tag{5.36}$$

$$\boldsymbol{\psi}_0 = \begin{pmatrix} \psi_0^{(1)} \\ \psi_0^{(2)} \end{pmatrix} = \int_0^\infty \begin{pmatrix} Q_0^{(1)}(x) \\ Q_0^{(2)}(x) \end{pmatrix} \frac{v(x)}{1-V(x)}dx, \tag{5.37}$$

$$\boldsymbol{\psi}_m = \begin{pmatrix} \psi_m^{(1)} \\ \psi_m^{(2)} \end{pmatrix} = \int_0^\infty \begin{pmatrix} P_{1,m}^{(1)}(x) \\ P_{1,m}^{(2)}(x) \end{pmatrix} \frac{h(x)}{1-S(x)}dx, \qquad (1 \le m \le M-1). \tag{5.38}$$

Using the above notations, (5.31), (5.32) and (5.33) are rewritten as

$$\boldsymbol{P}_1(z,0) = \boldsymbol{B}(z)\boldsymbol{Q}(z,0) - \boldsymbol{\psi}_0, \tag{5.39}$$

$$\boldsymbol{P}_m(z,0) = \boldsymbol{A}(z)\boldsymbol{P}_{m-1}(z,0) - \boldsymbol{\psi}_{m-1}, \qquad (2 \le m \le M), \tag{5.40}$$

$$\boldsymbol{Q}(z,0) = \boldsymbol{A}(z)\boldsymbol{P}_M(z,0) + \sum_{m=0}^{M-1}\boldsymbol{\psi}_m. \tag{5.41}$$

Also we define the following equations:

$$\widehat{\boldsymbol{Q}}(z,0) = \boldsymbol{B}(z)\boldsymbol{Q}(z,0), \tag{5.42}$$

$$\widehat{\boldsymbol{P}}_m(z,0) = \boldsymbol{A}(z)\boldsymbol{P}_m(z,0). \tag{5.43}$$

Note that

$$\hat{Q}(0,0) = \psi_0. \tag{5.44}$$

Therefore, from (5.39), (5.40) and (5.41), we obtain

$$\begin{aligned} P_1(z,0) &= \hat{Q}(z,0) - \psi_0, && (5.45)\\ P_m(z,0) &= \hat{P}_{m-1}(z,0) - \psi_{m-1}, \qquad (2 \le m \le M), && (5.46)\\ Q(z,0) &= \hat{P}_M(z,0) + \sum_{m=0}^{M-1} \psi_m. && (5.47) \end{aligned}$$

For simplicity, we define $A^0 = I$ where I is the identity matrix. It then follows from (5.45), (5.46) and (5.47) that

$$\begin{aligned} P_m(z,0) &= A^{m-1}(z)\hat{Q}(z,0) - \sum_{l=1}^{m} A^{l-1}(z)\psi_{m-l}, \qquad (1 \le m \le M), && (5.48)\\ Q(z,0) &= A^M(z)\hat{Q}(z,0) + \sum_{m=1}^{M} \left[I - A^m(z)\right]\psi_{M-m}. && (5.49) \end{aligned}$$

Multiplying both sides in (5.49) by $B(z)$ yields

$$\hat{Q}(z,0) = B(z)A^M(z)\hat{Q}(z,0) + \sum_{m=1}^{M} B(z)\left[I - A^m(z)\right]\psi_{M-m}. \tag{5.50}$$

The above equation becomes

$$\left[I - B(z)A^M(z)\right]\cdot\hat{Q}(z,0) = \sum_{m=1}^{M} B(z)\cdot\left[I - A^m(z)\right]\psi_{M-m}. \tag{5.51}$$

We define $\widehat{A}(z)$ and $\widehat{B}(z)$ as

$$\begin{aligned} \widehat{A}(z) &= z(1-\hat{p}(z)\hat{q}(z))A(z), && (5.52)\\ \widehat{B}(z) &= (1-\hat{p}(z)\hat{q}(z))B(z). && (5.53) \end{aligned}$$

Then, (5.51) can be rewritten as

$$\left[z^M(1-\hat{p}(z)\hat{q}(z))^{M+1}I - \widehat{B}(z)\widehat{A}^M(z)\right]\hat{Q}(z,0) = \sum_{m=1}^{M}\widehat{B}(z)\left[z^M(1-\hat{p}(z)\hat{q}(z))^M\{I - A^m(z)\}\right]\psi_{M-m}. \tag{5.54}$$

For abbreviation, $p(z)$, $q(z)$, $\hat{p}(z)$ and $\hat{q}(z)$ are described as p, q, $\hat{p}$ and $\hat{q}$, respectively. After some algebraic manipulations (see Appendix E.1 for details), we have

$$\left[z^M(1-\hat{p}\hat{q})^{M+1}I - \widehat{B}(z)\widehat{A}^M(z)\right]^{-1} = \frac{1}{(1-\hat{p}\hat{q})^{M+2}(z^M - f_M(p))(z^M - f_M(q))} \cdot \begin{pmatrix} z^M(1-\hat{p}\hat{q}) - f_M(q) + \hat{p}\hat{q}f_M(p) & \hat{q}(f_M(q) - f_M(p)) \\ \hat{p}(f_M(p) - f_M(q)) & z^M(1-\hat{p}\hat{q}) - f_M(p) + \hat{p}\hat{q}f_M(q) \end{pmatrix}, \tag{5.55}$$

where $f_m(z) = V^*(z)\{S^*(z)\}^m$. Then, $\widehat{Q}(z,0)$ is given by

$$\begin{aligned}\widehat{Q}(z,0) &= \sum_{m=1}^{M}\left[z^M(1-\hat{p}\hat{q})^{M+1}I - \widehat{B}(z)\widehat{A}^M(z)\right]^{-1}\widehat{B}(z) \\ &\qquad \cdot\left[z^M(1-\hat{p}\hat{q})^M(I-A^m(z))\right]\psi_{M-m} \\ &= \sum_{m=1}^{M}\frac{z^{M-m}}{(1-\hat{p}\hat{q})^2 a_p(z)a_q(z)}\cdot\begin{pmatrix} a_q(z)-\hat{p}\hat{q}a_p(z) & \hat{q}(a_p(z)-a_q(z)) \\ \hat{p}(a_q(z)-a_p(z)) & a_p(z)-\hat{p}\hat{q}a_q(z)\end{pmatrix} \\ &\quad\cdot\begin{pmatrix} V^*(p)b_p^{(m)}(z)-\hat{p}\hat{q}V^*(q)b_q^{(m)}(z) & \hat{q}(V^*(q)b_q^{(m)}(z)-V^*(p)b_p^{(m)}(z)) \\ \hat{p}(V^*(p)b_p^{(m)}(z)-V^*(q)b_q^{(m)}(z)) & V^*(q)b_q^{(m)}(z)-\hat{p}\hat{q}V^*(p)b_p^{(m)}(z)\end{pmatrix} \\ &\quad\cdot\psi_{M-m}, \end{aligned} \tag{5.56}$$

where

$$\begin{aligned} a_p(z) &= z^M - f_M(p), \qquad a_q(z) = z^M - f_M(q), \\ b_p^{(m)}(z) &= z^m - \{S^*(p)\}^m, \quad b_q^{(m)}(z) = z^m - \{S^*(q)\}^m. \end{aligned} \tag{5.57}$$

Thus, $Q(z,0)$ is expressed in terms of ψ_m $(0 \le m \le M-1)$ as

$$\begin{aligned} Q(z,0) &= B^{-1}(z)\widehat{Q}(z,0) \\ &= \frac{1}{(1-\hat{p}\hat{q})a_p(z)a_q(z)}\sum_{m=1}^{M} z^{M-m} \\ &\quad\cdot\begin{pmatrix} a_q(z)b_p^{(m)}(z)-\hat{p}\hat{q}a_p(z)b_q^{(m)}(z) & \hat{q}(a_p(z)b_q^{(m)}(z)-a_q(z)b_p^{(m)}(z)) \\ \hat{p}(a_q(z)b_p^{(m)}(z)-a_p(z)b_q^{(m)}(z)) & a_p(z)b_q^{(m)}(z)-\hat{p}\hat{q}a_q(z)b_p^{(m)}(z)\end{pmatrix} \\ &\quad\cdot\psi_{M-m}. \end{aligned} \tag{5.58}$$

Now, we consider the condition (5.11). Using (5.58), we have

$$\begin{aligned}\sum_{l=1}^{2}\int_0^\infty \overline{Q}^{(l)}(1,x)(1-V(x))dx &= E[V](Q^{(1)}(1,0)+Q^{(2)}(1,0)) \\ &= E[V]\sum_{k=0}^{M-1}\left[d_p^{(M-k)}(z)\right]_{z=1}(\psi_k^{(1)}+\psi_k^{(2)}), \end{aligned} \tag{5.59}$$

where

$$d_p^{(m)}(z) = \frac{d}{dz}b_p^{(m)}(z)\Big/\frac{d}{dz}a_p(z), \tag{5.60}$$

and

$$\left[d_p^{(m)}(z)\right]_{z=1} = \frac{m(1-\rho)}{M(1-\rho)-\lambda E[V]}. \tag{5.61}$$

Similarly, from (5.47),(5.48) and (5.56), we have

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{l=1}^{2}\int_0^\infty \overline{P}_m^{(l)}(1,x)(1-S(x))dx \\ &= E[S]\sum_{m=1}^{M}(P_m^{(1)}(1,0)+P_m^{(2)}(1,0)) \\ &= E[S]\sum_{k=0}^{M-1}\{M\left[d_p^{(M-k)}(z)\right]_{z=1}-(M-k)\}(\psi_k^{(1)}+\psi_k^{(2)}). \end{aligned} \tag{5.62}$$

Using (5.59) and (5.62), (5.11) is expressed in terms of $\psi_k^{(l)}$ $(0 \le k \le M-1,\ l = 1,2\ )$ as

$$\begin{aligned}
&\sum_{k=1}^{\infty}\sum_{m=1}^{M}\sum_{l=1}^{2}\int_0^{\infty} P_{k,m}^{(l)}(x)dx + \sum_{k=0}^{\infty}\sum_{l=1}^{2}\int_0^{\infty} Q_k^{(l)}(x)dx \\
&= E[V](Q^{(1)}(1,0) + Q^{(2)}(1,0)) + E[S]\sum_{m=1}^{M}(P_m^{(1)}(1,0) + P_m^{(2)}(1,0)) \\
&= \frac{E[V]}{M(1-\rho) - \lambda E[V]}\sum_{k=0}^{M-1}(M-k)(\psi_k^{(1)} + \psi_k^{(2)}) \\
&= 1. \qquad (5.63)
\end{aligned}$$

Again, we consider (5.58). We can easily show that $a_p(z) = 0$ and $a_q(z) = 0$ have M roots in a unit circle $|z| \le 1$ (see Appendix E.2). Let ω_i and θ_i $(0 \le i \le M-1)$ denote the roots of $a_p(z) = 0$ and $a_q(z) = 0$, respectively. We note that one real root, ω_0, of $a_p(z) = 0$ is 1. Both elements of (5.58), i.e. $Q^{(l)}(z,0)$'s $(l = 1,2\ )$, are analytical functions for $|z| \le 1$. Thus, the numerator of (5.58) should be zero for each $z = \omega_i$ $(1 \le i \le M-1)$ and $z = \theta_i$ $(0 \le i \le M-1)$. Therefore, substituting $z = \omega_i$ into (5.58), we obtain

$$\sum_{m=1}^{M}\omega_i^{M-m} b_p^{(m)}(\omega_i)\begin{pmatrix} 1 & -\hat{q}(\omega_i) \\ \hat{p}(\omega_i) & -\hat{p}(\omega_i)\hat{q}(\omega_i)\end{pmatrix}\boldsymbol{\psi}_{M-m} = \begin{pmatrix} 0 \\ 0\end{pmatrix}. \qquad (5.64)$$

The above equation becomes

$$\sum_{k=0}^{M-1}\omega_i^k b_p^{(M-k)}(\omega_i)(\psi_k^{(1)} - \hat{q}(\omega_i)\psi_k^{(2)}) = 0, \qquad (1 \le i \le M-1). \qquad (5.65)$$

Also we substitute $z = \theta_i$ into (5.58) and obtain

$$\sum_{k=0}^{M-1}\theta_i^k b_q^{(M-k)}(\theta_i)(-\hat{p}(\theta_i)\psi_k^{(1)} + \psi_k^{(2)}) = 0, \qquad (0 \le i \le M-1). \qquad (5.66)$$

(5.63), (5.65) and (5.66) are $2M$ independent and linear equations for $\psi_k^{(l)}$ $(0 \le k \le M-1,\ l = 1,2)$ (see Appendix E.3), so we can determine the value of $\psi_k^{(l)}$ from these equations.

5.3 Mean Queue Length and Waiting Time

In this section, we consider the mean queue length and the mean waiting time. We define the joint transforms $P_m^{*(l)}(z,s)$ $(1 \le m \le M,\ l = 1,2)$ and $Q^{*(l)}(z,s)$ $(l = 1,2)$ by

$$\begin{aligned}
P_m^{*(l)}(z,s) &= E[z^L e^{-s\widehat{S}}|\xi = m, \zeta = l]Prob\{\xi = m, \zeta = l\}, \quad (1 \le m \le M, l = 1,2), \qquad (5.67) \\
Q^{*(l)}(z,s) &= E[z^L e^{-s\widehat{V}}|\xi = 0, \zeta = l]Prob\{\xi = 0, \zeta = l\}, \quad (l = 1,2). \qquad (5.68)
\end{aligned}$$

Also we define the following vectors:

$$\begin{aligned}
\boldsymbol{P}_m^*(z,s) &= \begin{pmatrix} P_m^{*(1)}(z,s) \\ P_m^{*(2)}(z,s)\end{pmatrix}, \qquad (5.69) \\
\boldsymbol{Q}^*(z,s) &= \begin{pmatrix} Q^{*(1)}(z,s) \\ Q^{*(2)}(z,s)\end{pmatrix}. \qquad (5.70)
\end{aligned}$$

From (5.26), (5.30) and (5.36), the above equations become

$$\begin{aligned} \boldsymbol{Q}^*(z,s) &= \int_0^\infty e^{-sx}\boldsymbol{Q}(z,x)\{1-V(x)\}dx \\ &= \frac{1}{1-\hat{p}(z)\hat{q}(z)} \\ &\quad \cdot \begin{pmatrix} u(z,s)-\hat{p}(z)\hat{q}(z)w(z,s) & \hat{q}\{w(z,s)-u(z,s)\} \\ \hat{p}(z)\{u(z,s)-w(z,s)\} & w(z,s)-\hat{p}(z)\hat{q}(z)u(z,s) \end{pmatrix} \cdot \boldsymbol{Q}(z,0), \end{aligned} \tag{5.71}$$

$$\begin{aligned} \boldsymbol{P}_m^*(z,s) &= \int_0^\infty e^{-sx}\boldsymbol{P}_m(z,x)\{1-S(x)\}dx \\ &= \frac{1}{1-\hat{p}(z)\hat{q}(z)} \\ &\quad \cdot \begin{pmatrix} r(z,s)-\hat{p}(z)\hat{q}(z)t(z,s) & \hat{q}\{t(z,s)-r(z,s)\} \\ \hat{p}(z)\{r(z,s)-t(z,s)\} & t(z,s)-\hat{p}(z)\hat{q}(z)r(z,s) \end{pmatrix} \cdot \boldsymbol{P}_m(z,0), \\ &\qquad (1 \le m \le M), \end{aligned} \tag{5.72}$$

where

$$\begin{aligned} r(z,s) &= \frac{1-S^*(s+p(z))}{s+p(z)} \quad , \quad t(z,s) = \frac{1-S^*(s+q(z))}{s+q(z)}, \\ u(z,s) &= \frac{1-V^*(s+p(z))}{s+p(z)} \quad , \quad w(z,s) = \frac{1-V^*(s+q(z))}{s+q(z)}. \end{aligned} \tag{5.73}$$

Next, we define the joint transforms $P_m^*(z,s)$ $(1 \le m \le M)$ for the queue length and the elapsed service time, and $Q^*(z,s)$ for the queue length and the elapsed vacation time by

$$P_m^*(z,s) = E[z^L e^{-s\widehat{S}} | \xi = m] Prob\{\xi = m\}, \qquad (1 \le m \le M), \tag{5.74}$$

$$Q^*(z,s) = E[z^L e^{-s\widehat{V}} | \xi = 0] Prob\{\xi = 0\}. \tag{5.75}$$

Then, we have

$$\begin{aligned} \sum_{m=1}^{M} P_m^*(z,s) &= \sum_{m=1}^{M} \{P_m^{*(1)}(z,s) + P_m^{*(2)}(z,s)\} \\ &= \frac{z}{1-\hat{p}(z)\hat{q}(z)} \sum_{k=0}^{M-1} z^k \\ &\quad \cdot \Bigg\{ (1+\hat{p}(z)) r(z,s) \cdot \frac{b_p^{(M-k)}(z)}{b_p^{(1)}(z)} \cdot \frac{V^*(p(z))-1}{a_p(z)} \cdot (\psi_k^{(1)} - \hat{q}(z)\psi_k^{(2)}) \\ &\quad + (1+\hat{q}(z)) t(z,s) \cdot \frac{b_q^{(M-k)}(z)}{b_q^{(1)}(z)} \cdot \frac{V^*(q(z))-1}{a_q(z)} \cdot (-\hat{p}(z)\psi_k^{(1)} + \psi_k^{(2)}) \Bigg\}, \end{aligned} \tag{5.76}$$

$$\begin{aligned} Q^*(z,s) &= Q^{*(1)}(z,s) + Q^{*(2)}(z,s) \\ &= \frac{1}{1-\hat{p}(z)\hat{q}(z)} \sum_{k=0}^{M-1} z^k \Bigg\{ (1+\hat{p}(z)) u(z,s) \cdot \frac{b_p^{(M-k)}(z)}{a_p(z)} \cdot (\psi_k^{(1)} - \hat{q}(z)\psi_k^{(2)}) \\ &\quad + (1+\hat{q}(z)) w(z,s) \cdot \frac{b_q^{(M-k)}(z)}{a_q(z)} \cdot (-\hat{p}(z)\psi_k^{(1)} + \psi_k^{(2)}) \Bigg\}. \end{aligned} \tag{5.77}$$

Let $L(z)$ denote the generating function for the queue length at a random point in time. Using (5.76) and (5.77), we obtain $L(z)$ as

$$\begin{aligned} L(z) &= \lim_{s\to 0}\sum_{m=1}^{M} P_m^*(z,s) + \lim_{s\to 0} Q^*(z,s) \\ &= \frac{z-1}{1-\hat{p}(z)\hat{q}(z)}\sum_{k=0}^{M-1} z^k \\ &\quad \cdot\left\{(1+\hat{p}(z))\cdot\frac{1-V^*(p(z))}{p(z)}\cdot\frac{S^*(p(z))}{b_p^{(1)}(z)}\cdot\frac{b_p^{(M-k)}(z)}{a_p(z)}\cdot(\psi_k^{(1)}-\hat{q}(z)\psi_k^{(2)})\right. \\ &\quad \left.+(1+\hat{q}(z))\cdot\frac{1-V^*(q(z))}{q(z)}\cdot\frac{S^*(q(z))}{b_q^{(1)}(z)}\cdot\frac{b_q^{(M-k)}(z)}{a_q(z)}\cdot(-\hat{p}(z)\psi_k^{(1)}+\psi_k^{(2)})\right\}. \end{aligned} \tag{5.78}$$

Differentiating (5.78) and substituting $z=1$, the mean queue length $\overline{L}$ is found to be

$$\begin{aligned} \overline{L} &= \frac{1-\rho}{2}\cdot\frac{2M\rho-(M-1)\lambda E[V]}{M(1-\rho)-\lambda E[V]}+\frac{M(1-\rho)}{M(1-\rho)-\lambda E[V]}\cdot\frac{\lambda E[V^2]}{2E[V]} \\ &\quad +\frac{M\rho}{M(1-\rho)-\lambda E[V]}\cdot\frac{\lambda E[S^2]}{2E[S]}+\frac{ME[S]+E[V]}{M(1-\rho)-\lambda E[V]}\cdot\frac{\alpha\beta}{\alpha+\beta}\left(\frac{\lambda_1-\lambda_2}{\alpha+\beta}\right)^2 \\ &\quad +\left(\sum_{k=0}^{M-1}\frac{(M-k)E[V]}{M(1-\rho)-\lambda E[V]}\cdot\psi_k^{(2)}-\frac{\alpha}{\alpha+\beta}\right)\cdot\frac{\lambda_1-\lambda_2}{\alpha+\beta} \\ &\quad +\frac{1-\rho}{2}\cdot\frac{E[V]}{M(1-\rho)-\lambda E[V]}\sum_{k=0}^{M-1}(M-k)k(\psi_k^{(1)}+\psi_k^{(2)}). \end{aligned} \tag{5.79}$$

From Little's formula, we obtain the mean waiting time $\overline{W}$ as

$$\overline{W} = \overline{L}/\lambda. \tag{5.80}$$

5.4 Numerical Results

In this section, we show some numerical examples of the results obtained in Section 5.3. The service time and the vacation time distributions are chosen as follows.

- Service time S is exponentially distributed and its mean is 1.0.
- Vacation time V is constant (= 1.0).

Fig.5.1 illustrates the mean waiting time for various values of the limit number M as a function of the overall arrival rate λ. We set $\lambda_1 : \lambda_2 = 2:1$ and $\alpha = \beta = 0.2$. We observe that the mean waiting time tends to infinity as the increase of λ in each case. Also we observe that when λ approaches the values $M/(ME[S]+E[V])$, the mean waiting time increases suddenly.

Fig.5.2 illustrates the mean waiting time for various values of the parameter u as a function of λ, where $\lambda_1 : \lambda_2$ is equal to $u:1$. We set $M=5$ and $\alpha=\beta=0.2$. Note that the arrival process is a Poisson process when $u=1$. The mean waiting time becomes large when the value of u increases. This shows that the mean waiting time is affected by the ratio of two arrival rates even when the overall arrival rate λ is fixed.

Fig.5.3 shows the effect of the mean sojourn time in each state on the mean waiting time for various values of u. We set $M = 5$, $\alpha = \beta$ and $\lambda = 0.8$. Note that in the case of $u = 1$, the mean waiting time is constant regardless of the mean sojourn time. We observe that the mean waiting time becomes large when the value of u increases. From this observation, it turns out that the mean waiting time is affected strongly by the arrival rate and the mean sojourn time in each state of the arrival process.

5.5 Conclusion

In this chapter, we consider an $SPP/G/1$ system with multiple vacations and E-limited service discipline. Using the supplementary variable technique, we derive the transform of the stationary queue length distribution explicitly. Numerical results show that the mean waiting time is affected by the limit size M, the arrival rate and the sojourn time in each state of the arrival process.

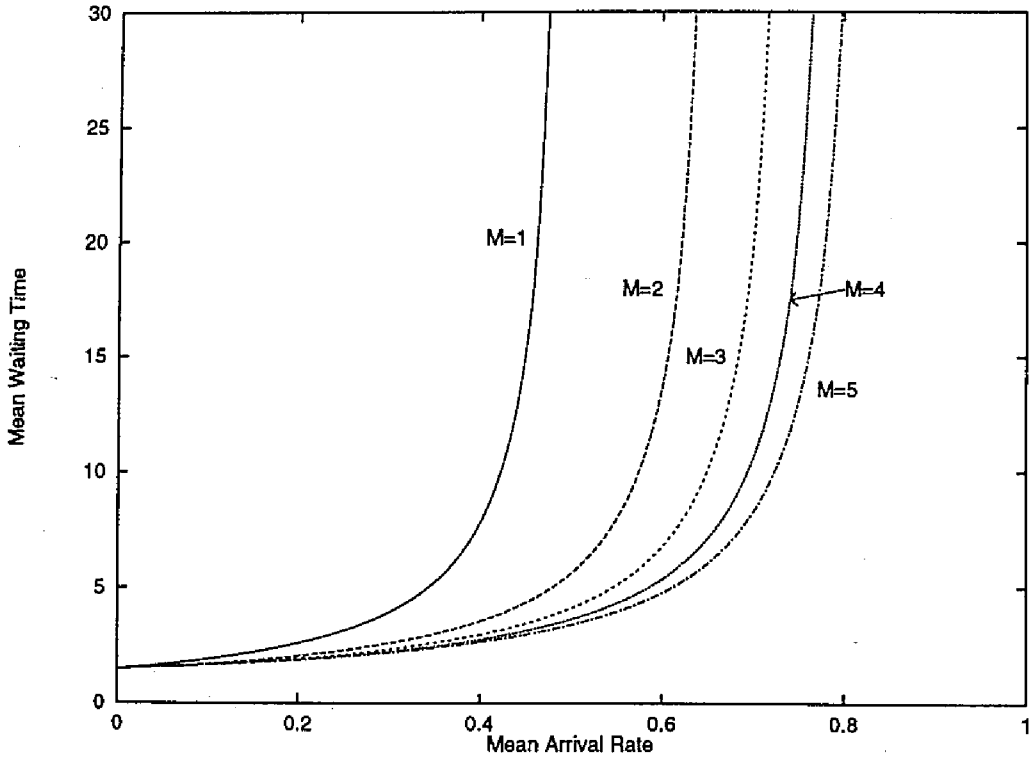


Figure 5.1: Mean Waiting Time of an $SPP/G/1$ System

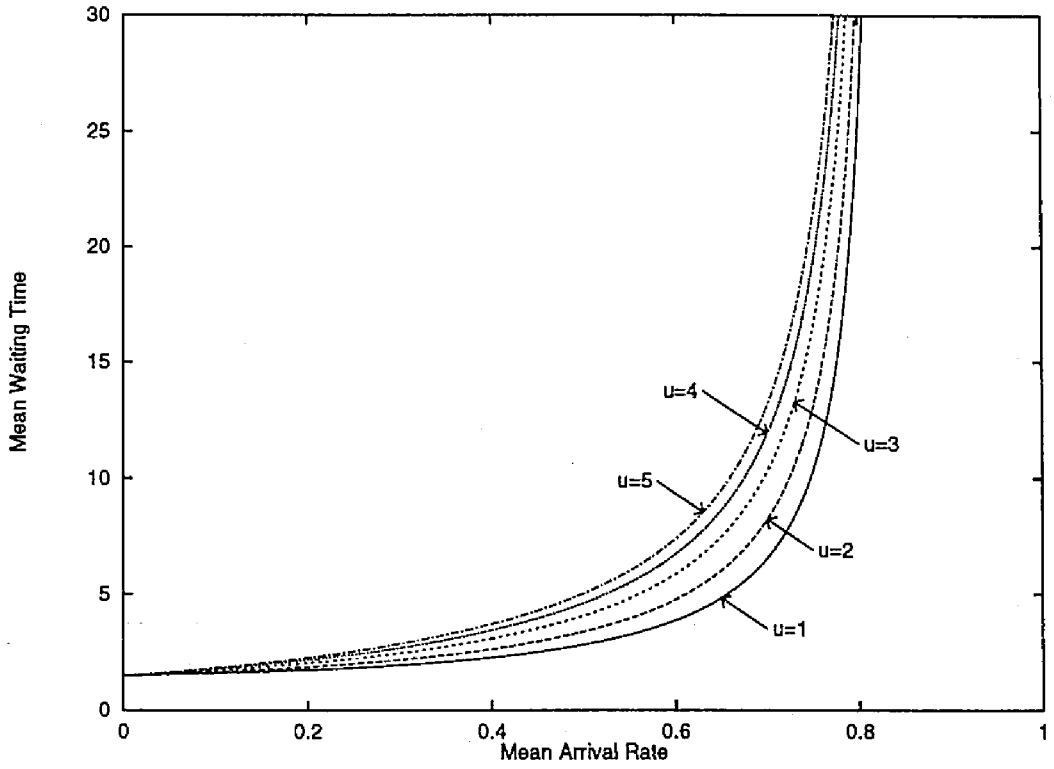


Figure 5.2: Mean Waiting Time of an $SPP/G/1$ System

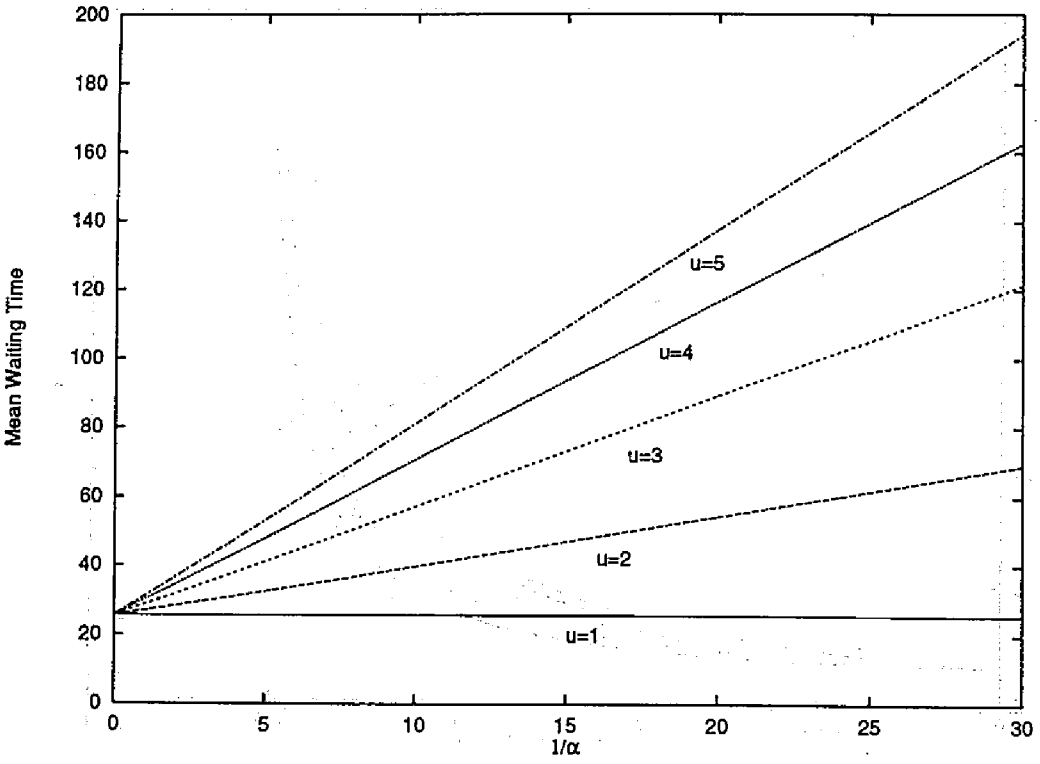


Figure 5.3: Mean Waiting Time of an $SPP/G/1$ System

Chapter 6

MAP/G/1 Queues under N-policy with and without Vacations

6.1 Introduction

In Chapter 5, we analyzed the $SPP/G/1$ system where the arrival process SPP is the special case of the $MMPP$. In this chapter, we consider the queueing systems with MAP[1], the fairly extended arrival process. The MAP includes as special cases the $MMPP$ and the superposition of phase-type renewal processes. Asmussen and Koole [Asmu93] have also shown that MAP is weakly dense in the class of stationary simple point processes. Therefore MAP is a fairly general process and has a capability of representing a wide class of arrival processes.

In this chapter, we consider $MAP/G/1$ vacation models with the following characteristics. Messages arrive to the system according to a MAP with representation (C, D), where C and D are $m \times m$ matrices. Note that m denotes the number of phases in the underlying Markov chain which governs the arrival process. Service times are i.i.d. according to a general PDF $S(x)$ with finite mean $E[S]$, whose LST is denoted by $S^*(s)$. As for the vacation policy, we consider the following two situations:

1. N-policy without vacations
   At the end of a busy period, the server is turned off and inspects the queue length every time a message arrives. When the queue length reaches a pre-specified value N, the server turns on and serves messages continuously until the system becomes empty.

2. N-policy with vacations
   At the end of a busy period, the server takes a sequence of vacations, where vacation times are i.i.d. according to a general PDF $V(x)$ with finite mean $E[V]$. At the end of each vacation, the server inspects the queue length. If the queue length is greater than or equal to a pre-specified value N at this time, the server begins to serve messages continuously until the system becomes empty.

In both cases, there is a possibility that the server remains being idle even when some messages are waiting for their services. Thus, both queues with the above features fall into a category of queues with generalized vacations [Fuhr85]. Note that when $N = 1$ without vacations, our queueing model is reduced to the ordinary $MAP/G/1$ queue. Also when $N = 1$ with vacations, our queueing model is reduced to the $MAP/G/1$ with multiple vacations and

[1] We summarized some properties of the MAP in section 1.3 of Chapter 1.

the exhaustive service. Thus, the queueing models considered in this chapter are regarded as generalizations of those which have been analyzed.

The queueing system under N-policy without vacations has been one of the classical subjects on control of queues (see [Heym82] and references therein). As for the N-policy with vacations, there also have been a number of works. Among them, Hofri [Hofi86] and Kella [Kell89] studied the same control policy for the $M/G/1$ system. Lee and Srinivasan [Lee89b] studied the $M^X/G/1$ system under the N-policy with vacations.

A typical application for N-policy is the quality control problem [Kell89]. A manufacturing plant produces certain items that occasionally are defective. The good items are marketed while the defective ones are kept in storage until they can be reworked to meet specifications. Assume that one of the machines in the plant may be converted as needed from production mode to a repair mode in order to perform this rework. The question is what would be an appropriate cutoff number N such that if the number of defective items is at least N, then the special machine will be converted from the production mode to the repair mode at the next opportunity. After conversion to repair mode, this machine will rework all of the defective items (including new arrivals) exhaustively, and then switch back to the production mode when there are no defective items left.

We can interpret the defective items as the served customer and the special machine as the server, where this server is available for serving these customers only when the machine is in the repair mode. The service time is the time required to rework a defective item to meet specifications.

If we count the number of defectives at each time when the defective is produced, we then have a queueing system under N-policy without vacations. On the other hand, if we inspect the number of defectives after a certain period, we have a queueing system under N-policy with vacations.

In [Kell89], authors assumed that defective items occur according to a Bernoulli trial for each machine, and hence, the superposition of the output processes of defective items from the various machines could be regarded as a Poisson process. However, if we consider a few production machines, the MAP is suitable for modeling the arrival process.

The queueing models considered in this chapter are formulated as Markov chains of $M/G/1$ type [Neut89]. However, the boundary behavior in our queueing models is complicated, especially in the N-policy with vacations. Thus, the usual approach given in [Neut89] does not seems to be efficient. We provide an alternative approach to compute an essential quantity related to the boundary behavior. Thus, combined with the established methods in [Luca90], [Neut89] and [Taki93b], this approach gives a simple and efficient algorithm to compute various quantities of interest.

The remainder of this chapter is organized as follows. In section 6.2, we study the queue length and waiting time distributions for N-policy without vacations. We derive the recursive formulas to compute the queue length distribution, the factorial moments of the queue length distribution and the moments of the actual waiting time distribution. In section 6.3, we study the queue length and actual waiting time distributions for N-policy with vacations. We derive the recursive formulas to compute the queue length distribution, its factorial moments and the moments of the waiting time distribution. In section 6.4, we show some numerical examples using the moment formulas of the waiting time for N-policy with and without vacations. In particular, we show that in light traffic, the correlation in arrivals leads to a smaller mean waiting time. Throughout the chapter, we assume that the system is in equilibrium.

6.2 N-policy without Vacations

In this section, we consider a $MAP/G/1$ queue under N-policy without vacations in equilibrium. First, we consider the stationary queue length at departures. Then, we consider the stationary queue length distribution at an arbitrary time. We also derive the LST of the actual waiting time distribution for an arriving message.

6.2.1 Generating Function for Queue Length at Departures

We consider the imbedded Markov chain at departure epochs. Let A_n $(n \geq 0)$ denote an $m \times m$ matrix whose (i,j)th element represents the conditional probability that n messages arrive to the system during a service time of a message and the underlying Markov chain is in phase j at the end of the service given that the underlying Markov chain is in phase i at the beginning of the service. In the queue under N-policy without vacations, the transition probability matrix P is given by

$$P = \begin{bmatrix} O & O & O & \cdots & O & B_{N-1} & B_N & \cdots \\ A_0 & A_1 & A_2 & \cdots & A_{N-2} & A_{N-1} & A_N & \cdots \\ O & A_0 & A_1 & \cdots & A_{N-3} & A_{N-2} & A_{N-1} & \cdots \\ O & O & A_0 & \cdots & A_{N-4} & A_{N-3} & A_{N-2} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \\ O & O & O & \cdots & A_0 & A_1 & A_2 & \cdots \\ O & O & O & \cdots & O & A_0 & A_1 & \cdots \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \end{bmatrix}, \tag{6.1}$$

where B_n $(n \geq N-1)$ denotes an $m \times m$ matrix which is given by

$$B_n = \left[(-C)^{-1}D\right]^N A_{n-N+1}, \qquad n \geq N-1.$$

Note that the factor $(-C)^{-1}D$ represents the phase transition matrix during an interarrival time [Luca90]. As for the computation of A_n, readers are referred to [Taki93b]. Let $A(z)$ and $B(z)$ denote matrix generating functions of the A_n and the B_n, respectively:

$$A(z) = \sum_{n=0}^{\infty} A_n z^n, \qquad B(z) = \sum_{n=N-1}^{\infty} B_n z^n. \tag{6.2}$$

We then have [Luca90]

$$A(z) = \int_0^{\infty} e^{(C+zD)x} dS(x). \tag{6.3}$$

Furthermore, $B(z)$ is given in terms of $A(z)$:

$$B(z) = \left[(-C)^{-1} Dz\right]^N \frac{A(z)}{z}.$$

Let x_k $(k \geq 0)$ denote a $1 \times m$ vector whose ith element represents the stationary joint probability that the number of messages in the system at departures is k and the phase of the arrival process is i. Furthermore, we define the vector generating function $\boldsymbol{X}(z)$ as

$$\boldsymbol{X}(z) = \sum_{k=0}^{\infty} x_k z^k.$$

From (6.1), we have the following equation:

$$X(z) = x_0 B(z) + [X(z) - x_0] \frac{A(z)}{z}, \tag{6.4}$$

from which we obtain

$$X(z)[zI - A(z)] = x_0 \left\{ \left[(-C)^{-1} D \right]^N z^N - I \right\} A(z). \tag{6.5}$$

Thus, once we obtain x_0, the vector generating function $X(z)$ is completely determined. Before considering x_0, we derive some formulas which will be used later.

Let π denote a $1 \times m$ vector whose ith element represents the stationary probability of the underlying Markov chain being phase i. Note that π satisfies

$$\pi(C + D) = 0, \qquad \pi e = 1, \tag{6.6}$$

where e denotes an $m \times 1$ vector whose all elements are equal to one. Setting $z = 1$ in (6.5) and adding $X(1)e\pi$ to both sides yield

$$X(1) = \pi + x_0 \left\{ \left[(-C)^{-1} D \right]^N - I \right\} A(I - A + e\pi)^{-1}, \tag{6.7}$$

where $A = A(1)$. We define β as $\beta = A'(1)e$. Post-multiplying both sides of (6.7) by β, we obtain

$$X(1)\beta = \rho + x_0 \left\{ \left[(-C)^{-1} D \right]^N - I \right\} A(e\pi - C - D)^{-1} De, \tag{6.8}$$

where ρ denotes the utilization factor which is given by $\pi\beta$. Due to the assumption that the system is in equilibrium, we have $\rho < 1$. In the derivation of (6.8), we use the equality

$$(I - A + e\pi)^{-1}\beta = (e\pi - C - D)^{-1} De + (\rho - \pi De)e,$$

which comes from (6.3) and (6.6).

On the other hand, differentiating (6.5) with respect to z, setting $z = 1$ and post-multiplying both sides by e yield

$$1 - X(1)\beta = N x_0 e + x_0 \left\{ \left[(-C)^{-1} D \right]^N - I \right\} (I - A)(e\pi - C - D)^{-1} De, \tag{6.9}$$

where we use the equality

$$\beta = (I - A)(e\pi - C - D)^{-1} De + \rho e,$$

which again comes from (6.3) and (6.6). From (6.8) and (6.9), we obtain

$$\begin{aligned} 1 - \rho &= N x_0 e + x_0 \left\{ \left[(-C)^{-1} D \right]^N - I \right\} (e\pi - C - D)^{-1} De \qquad (6.10) \\ &= N x_0 e + x_0 \sum_{k=0}^{N-1} \left[(-C)^{-1} D \right]^k (-C)^{-1} (C + D)(e\pi - C - D)^{-1} De \\ &= \lambda x_0 \sum_{k=0}^{N-1} \left[(-C)^{-1} D \right]^k (-C)^{-1} e, \end{aligned}$$

where λ denotes the mean arrival rate which is given by πDe. Note that $\rho = \lambda E[S]$, which can be verified with (6.3).

Remarks. (6.10) can be rewritten to be

$$\rho = E[S] \Big/ \left\{ E[S] + x_0 e \frac{x_0}{x_0 e} \sum_{k=0}^{N-1} \left[(-C)^{-1}D\right]^k (-C)^{-1}e \right\},$$

where the right hand side is considered as a time fraction of the server being busy between consecutive imbedded points.

6.2.2 Determination of the Vector x_0

In this subsection, we obtain a formula to compute x_0. We define the level i as the set of states $\{(i, 1), \cdots, (i, m)\}$, $i \geq 0$. We first consider the state transition of the underlying Markov chain during the first passage time from level $i+1$ to level i ($i \geq 0$). Let G denote an $m \times m$ matrix which represents the state transition matrix of the underlying Markov chain during the first passage time. Then we have [Neut89]

$$G = \sum_{\nu=0}^{\infty} A_\nu G^\nu. \tag{6.11}$$

Note that G is stochastic when $\rho < 1$. Also G satisfies the following equation [Luca90]:

$$G = \int_0^\infty e^{(C+DG)x} dS(x).$$

As for the computation of G, readers are referred to [Luca90] and [Taki93b].

Using G, we consider the state transition of the underlying Markov chain during the recurrence time of the level 0. Let K denote an $m \times m$ matrix which represents the state transition matrix of the underlying Markov chain during the recurrence time. Note that K satisfies

$$K = \left[(-C)^{-1}D\right]^N G^N. \tag{6.12}$$

Let κ denote the invariant probability vector of K, which satisfies $\kappa K = \kappa$ and $\kappa e = 1$. Once we obtain κ, we can readily obtain x_0. Let $\overline{K}$ denote the mean recurrence time of level zero. By definition, x_0 is given in terms of κ and $\overline{K}$ [Neut89]

$$x_0 = \frac{\kappa}{\overline{K}}. \tag{6.13}$$

Substituting x_0 in (6.13) into (6.10), and solving with respect to $\overline{K}$, we have

$$\overline{K} = \frac{\lambda}{1-\rho} \kappa \sum_{k=0}^{N-1} \left[(-C)^{-1}D\right]^k (-C)^{-1}e.$$

Thus, $\overline{K}$ is given in terms of κ and the vector x_0 is given by (6.13).

Remarks. In the ordinary $M/G/1$ paradigm, we first compute the invariant probability vector g of G, and then obtain κ and $\overline{K}$ in terms of g [Neut89]. However, in our formulation, we derive the quantities of interest only in terms of κ and we don't need to compute g.

6.2.3 Queue Length Distribution at Departure and its Moments

In this subsection, we provide the computational algorithm for the queue length distribution $\boldsymbol{x}_k$ ($k \geq 1$) at departures and its moments. Note that a stable algorithm for the Markov chain of $M/G/1$ type is provided in [Rama88]. Since (6.1) is of $M/G/1$ type [Neut89], we follows the algorithm in [Rama88] and obtain the following recursion for $\boldsymbol{x}_k$ ($k \geq 1$):

$$\boldsymbol{x}_k = \left[\boldsymbol{x}_0 \overline{B}_k + \sum_{j=1}^{k-1} \boldsymbol{x}_j \overline{A}_{k+1-j} \right] \left(I - \overline{A}_1 \right)^{-1},$$

where

$$\overline{A}_k = \sum_{n=k}^{\infty} A_n G^{n-k}, \qquad k \geq 1, \tag{6.14}$$

$$\overline{B}_k = \sum_{n=N-1}^{\infty} B_n G^{n-k}, \qquad 1 \leq k \leq N-2, \tag{6.15}$$

$$\overline{B}_k = \sum_{n=k}^{\infty} B_n G^{n-k}, \qquad k \geq N-1. \tag{6.16}$$

Next we provide a recursive formula to compute the factorial moments of the queue length distribution at departures. We define $\boldsymbol{X}^{(n)}$, $A^{(n)}$ and $B^{(n)}$ as

$$\boldsymbol{X}^{(n)} = \lim_{z \to 1} \frac{d^n}{dz^n} \boldsymbol{X}(z), \qquad A^{(n)} = \lim_{z \to 1} \frac{d^n}{dz^n} A(z), \qquad B^{(n)} = \lim_{z \to 1} \frac{d^n}{dz^n} B(z).$$

We then follow the approach in [Neut89] and obtain the following recursion for the factorial moments of queue length distribution at departures:

$$\boldsymbol{U}^{(n)} = \begin{cases} \boldsymbol{x}_0 (B(1) - A(1)), & n = 0, \\ \boldsymbol{x}_0 (B^{(1)} + B^{(0)} - A^{(1)}), & n = 1, \\ \displaystyle\sum_{m=0}^{n-2} \binom{n}{m} \boldsymbol{X}^{(m)} A^{(n-m)} + \boldsymbol{x}_0 (B^{(n)} + n B^{(n-1)} - A^{(n)}), & n \geq 2, \end{cases}$$

$$\boldsymbol{X}^{(n)} \boldsymbol{e} = \frac{\boldsymbol{U}^{(n+1)} \boldsymbol{e}}{(n+1)(1-\rho)} + \frac{1}{1-\rho} \{ \boldsymbol{U}^{(n)} - n \boldsymbol{X}^{(n-1)} (I - A^{(1)}) \} \cdot [I - A(1) + \boldsymbol{e}\boldsymbol{\pi}]^{-1} A^{(1)} \boldsymbol{e}, \qquad n \geq 1,$$

$$\boldsymbol{X}^{(n)} = \begin{cases} \boldsymbol{\pi} + \boldsymbol{U}^{(0)} [I - A(1) + \boldsymbol{e}\boldsymbol{\pi}]^{-1}, & n = 0, \\ \boldsymbol{X}^{(n)} \boldsymbol{e}\boldsymbol{\pi} + \{ \boldsymbol{U}^{(n)} - n \boldsymbol{X}^{(n-1)} (I - A^{(1)}) \} [I - A(1) + \boldsymbol{e}\boldsymbol{\pi}]^{-1}, & n \geq 1, \end{cases}$$

where

$$\boldsymbol{X}^{(0)} = \boldsymbol{X}(1), \qquad A^{(0)} = A(1), \qquad B^{(0)} = B(1).$$

Namely, computing $\boldsymbol{U}^{(0)}$, $\boldsymbol{X}^{(0)}$, $\boldsymbol{U}^{(1)}$ and then $\boldsymbol{U}^{(k+1)}$, $\boldsymbol{X}^{(k)} \boldsymbol{e}$, $\boldsymbol{X}^{(k)}$ in this order, we obtain the nth factorial moment $\boldsymbol{X}^{(n)}$ of the queue length distribution at departures.

6.2.4 Queue Length Distribution at an Arbitrary Time and its Moments

In this subsection, we consider the distribution of the number of messages in the system at an arbitrary time. Let y_n denote a $1 \times m$ vector whose ith element is the stationary joint probability that the number of messages in the system is n and the phase of the arrival process is i at an arbitrary time. We define the vector generating function $\boldsymbol{Y}(z)$ as

$$\boldsymbol{Y}(z) = \sum_{n=0}^{\infty} y_n z^n.$$

$\boldsymbol{Y}(z)$ consists of the idle term and busy one. Let U denote the idle time of the server. Then, we obtain the mean idle time as

$$E[U] = \frac{\boldsymbol{x}_0}{\boldsymbol{x}_0 \boldsymbol{e}} \sum_{k=0}^{N-1} \left[(-C)^{-1}D\right]^k (-C)^{-1}\boldsymbol{e}. \tag{6.17}$$

Using (6.10) and (6.17), we obtain the vector whose ith element represents the conditional probability that the number of messages in the system is k and the phase of the arrival process is i given that the server is idle:

$$\frac{1}{E[U]}\frac{\boldsymbol{x}_0}{\boldsymbol{x}_0 \boldsymbol{e}} \left[(-C)^{-1}D\right]^k (-C)^{-1} = \frac{\lambda}{1-\rho}\boldsymbol{x}_0 \left[(-C)^{-1}D\right]^k (-C)^{-1}. \tag{6.18}$$

Using (6.18), we obtain

$$\begin{aligned} \boldsymbol{Y}(z) &= (1-\rho)\sum_{k=0}^{N-1} \frac{\lambda}{1-\rho}\boldsymbol{x}_0 \left[(-C)^{-1}D\right]^k (-C)^{-1} z^k \\ &\quad +\rho\left\{\boldsymbol{X}(z) - \boldsymbol{x}_0 + \boldsymbol{x}_0 \left[(-C)^{-1}D\right]^N z^N\right\} A^*(z) \\ &= \lambda \boldsymbol{x}_0 \sum_{k=0}^{N-1} \left[(-C)^{-1}D\right]^k (-C)^{-1} z^k \\ &\quad +\rho\left\{\boldsymbol{X}(z) - \boldsymbol{x}_0 + \boldsymbol{x}_0 \left[(-C)^{-1}D\right]^N z^N\right\} A^*(z), \end{aligned} \tag{6.19}$$

where $A^*(z)$ is the matrix generating function of the number of arrivals during the forward recurrence time of a service time and given by [Luca90]

$$A^*(z) = \frac{1}{E[S]}\left[A(z) - I\right](C + zD)^{-1}. \tag{6.20}$$

From (6.4) and (6.20), the second term in (6.19) becomes

$$\begin{aligned} &\rho\left\{\boldsymbol{X}(z) - \boldsymbol{x}_0 + \boldsymbol{x}_0 \left[(-C)^{-1}D\right]^N z^N\right\} A^*(z) \\ &\quad = \lambda(z-1)\boldsymbol{X}(z)(C+zD)^{-1} - \lambda \boldsymbol{x}_0 \sum_{k=0}^{N-1} \left[(-C)^{-1}D\right]^k (-C)^{-1} z^k. \end{aligned} \tag{6.21}$$

Substituting (6.21) into (6.19), we obtain

$$\boldsymbol{Y}(z) = \lambda(z-1)\boldsymbol{X}(z)(C+zD)^{-1}. \tag{6.22}$$

(6.22) shows the relationship between the queue length distribution at departures and at an arbitrary time. Since this relationship holds for any stationary queue with MAP arrivals [Taki94a], an independent verification provides a validation for our analysis so far.

Post-multiply both sides of (6.22) by $(C + zD)$ and comparing the coefficients of z^k in both sides, we obtain the following recursion for y_k $(k \geq 0)$ in terms of the x_k:

$$y_0 = \lambda x_0(-C)^{-1}, \tag{6.23}$$

$$y_k = y_{k-1}D(-C)^{-1} + \lambda(x_k - x_{k-1})(-C)^{-1}, \qquad k \geq 1. \tag{6.24}$$

Next we consider the factorial moments of the queue length distribution at an arbitrary time. We define $Y^{(n)}$ as

$$Y^{(n)} = \lim_{z \to 1} \frac{d^n}{dz^n} Y(z).$$

We follow the approach in [Luca90] and obtain the following recursion to compute $Y^{(n)}$ $(n \geq 1)$:

$$\begin{aligned} Y^{(0)} &= \pi, \\ Y^{(n)}e &= X^{(n)}e - n\left(Y^{(n-1)}D/\lambda - X^{(n-1)}\right)(e\pi - C - D)^{-1}De, \quad n \geq 1, \\ Y^{(n)} &= Y^{(n)}e\pi + n\left(Y^{(n-1)}D - \lambda X^{(n-1)}\right)(e\pi - C - D)^{-1}, \quad n \geq 1, \end{aligned}$$

where $Y^{(0)} = Y(1)$.

6.2.5 LST for Actual Waiting Time and its Moments

In this section, we consider the waiting time distribution of an arriving message. To do so, we first consider the waiting time of a message which arrives when the server is idle. Let y_k^+ denote a $1 \times m$ vector whose ith element represents the joint probability that a message arrives when the server is idle, finds k waiting messages upon arrival, and the state of the arrival process immediately after the arrival is i. Using (6.18), We then have

$$y_k^+ = (1-\rho) \cdot \frac{\lambda}{1-\rho} x_0 \left[(-C)^{-1}D\right]^k (-C)^{-1}D/\lambda = x_0 \left[(-C)^{-1}D\right]^{k+1}.$$

Thus, the LST $W_1^*(s)$ of the waiting time distribution when the message arrives during an idle time of the server is given by

$$\begin{aligned} W_1^*(s) &= \sum_{k=0}^{N-1} y_k^+ [(sI - C)^{-1}D]^{N-k-1} e[S^*(s)]^k \\ &= x_0 \sum_{k=0}^{N-1} [(-C)^{-1}D]^{k+1} \left[(sI - C)^{-1}D\right]^{N-k-1} [S^*(s)]^k e. \end{aligned}$$

Next, we consider the waiting time of a message which arrives when the server is busy. To do so, we first derive the joint transform for the number of messages and the forward recurrence time of the current service when the server is busy. Note that the server is busy with probability ρ. Given that the server is busy, messages in the system is classified into two types. One includes messages which are in the system when the current service starts. The other includes messages which arrive during the backward recurrence time of the current service. Thus we have the joint transform $Y^*(z, s)$ for the number of messages and the forward recurrence time at an arbitrary point of the current service:

$$Y^*(z,s) = \rho\left\{\boldsymbol{X}(z) - x_0 + z^N x_0[(-C)^{-1}D]^N\right\} A(z,s),$$

where $A(z,s)$ denotes the joint transformed matrix for the number of messages which arrive in the backward recurrence time and the forward recurrence time, and is given by

$$\begin{aligned} A(z,s) &= \int_0^\infty \frac{x dS(x)}{E[S]} \int_0^x \frac{dt}{x} e^{(C+zD)t} e^{-s(x-t)} \\ &= \frac{A(z) - S^*(s)I}{E[S]} [sI + C + zD]^{-1}. \end{aligned}$$

Therefore we obtain the LST $W_2^*(s)$ for the waiting time distribution of a message which arrives when the server is busy as follows:

$$\begin{aligned} W_2^*(s) &= Y^*(S^*(s), s) De/\lambda S^*(s) \\ &= \boldsymbol{x}_0 \left\{ I - [(-C)^{-1}D]^N [S^*(s)]^N \right\} [sI + C + S^*(s)D]^{-1} De, \end{aligned}$$

where we use the equality

$$\boldsymbol{X}(S^*(s))[S^*(s)I - A(S^*(s))] = \left\{ \boldsymbol{x}_0 [(-C)^{-1}D]^N [S^*(s)]^N - I \right\} A(S^*(s)),$$

which comes from (6.5).

Let $W^*(s)$ denote the LST for the actual waiting time distribution. By definition, $W^*(s)$ is given by $W_1^*(s) + W_2^*(s)$. Therefore we obtain

$$\begin{aligned} W^*(s) &= \boldsymbol{x}_0 \sum_{k=0}^{N-1} [(-C)^{-1}D]^{k+1} \left[(sI - C)^{-1}D \right]^{N-k-1} [S^*(s)]^k e \\ &\quad + \boldsymbol{x}_0 \left\{ I - [(-C)^{-1}D]^N [S^*(s)]^N \right\} [sI + C + S^*(s)D]^{-1} De. \end{aligned} \tag{6.25}$$

We now consider the moments of the actual waiting time. We first define $W^{(n)}$ as

$$W^{(n)} = \lim_{s \to 0} (-1)^n \frac{d^n}{ds^n} W(s), \qquad n \geq 1.$$

To obtain the recursive formula to compute $W^{(n)}$, We rewrite (6.25) as

$$W^*(s) = \boldsymbol{x}_0 \sum_{k=0}^{N-1} [(-C)^{-1}D]^{k+1} T_k(s) e + \boldsymbol{x}_0 T(s) De,$$

where

$$\begin{aligned} T_k(s) &= \left[(sI - C)^{-1}D \right]^{N-k-1} [S^*(s)]^k, \qquad 0 \leq k \leq N-1, \\ T(s) &= \left\{ I - [(-C)^{-1}D]^N [S^*(s)]^N \right\} [sI + C + S^*(s)D]^{-1}. \end{aligned} \tag{6.26}$$

We then have

$$W^{(n)} = \boldsymbol{x}_0 \sum_{k=0}^{N-1} [(-C)^{-1}D]^{k+1} T_k^{(n)} e + \boldsymbol{x}_0 T^{(n)} De, \qquad n \geq 1,$$

where for $n \geq 1$,

$$T_k^{(n)} = \lim_{s \to 0} (-1)^n \frac{d^n}{ds^n} T_k(s), \qquad T^{(n)} = \lim_{s \to 0} (-1)^n \frac{d^n}{ds^n} T(s).$$

Thus once we have $T_k^{(n)}$ and $T^{(n)}$, $W^{(n)}$ is readily obtained. In what follows, we provide the recursive formula to compute $T_k^{(n)}$ and $T^{(n)}$.

First, we consider $T_k^{(n)}$ $(n \geq 1)$. We define $H_k(s)$ and $S_k(s)$ as

$$H_k(s) = \left[(sI - C)^{-1}D\right]^k, \qquad S_k(s) = [S^*(s)]^k.$$

Then, $T_k(s) = H_{N-k-1}(s)S_k(s)$. Furthermore, we define $H_k^{(n)}$ and $S_k^{(n)}$ as

$$H_k^{(n)} = \lim_{s\to 0}(-1)^n \frac{d^n}{ds^n} H_k(s), \qquad S_k^{(n)} = \lim_{s\to 0}(-1)^n \frac{d^n}{ds^n} S_k(s),$$

and $S^{(n)} = S_1^{(n)}$. Note that $S^{(1)} = E[S]$. Taking the nth derivative of $H_1(s)$, we obtain $H_1^{(n)} = n!\ (-C)^{-(n+1)}D$. Since $H_k(s) = H_1(s)\ H_{k-1}(s)$, we compute the nth derivative $H_k^{(n)}$ using the recursion

$$H_k^{(n)} = \sum_{i=0}^{n} \binom{n}{i} H_1^{(i)}\ H_{k-1}^{(n-i)},$$

where $H_k^{(0)} = [(-C)^{-1}D]^k$. Similarly, we compute the nth derivative $S_k^{(n)}$ using the recursion

$$S_k^{(n)} = \sum_{i=0}^{n} \binom{n}{i} S^{(i)}\ S_{k-1}^{(n-i)},$$

where $S^{(0)} = 1$. Thus we obtain the nth derivative $T_k^{(n)}$ by

$$T_k^{(n)} = \sum_{i=0}^{n} \binom{n}{i} H_{N-k-1}^{(i)}\ S_k^{(n-i)}.$$

Secondly, we consider the nth derivative $T^{(n)}$ of $T(s)$. Using (6.26), it follows

$$T(s)[sI + C + S^*(s)D] = U(s),$$

where

$$U(s) = I - [(-C)^{-1}D]^N [S^*(s)]^N.$$

We define $U^{(n)}$ $(n \geq 1)$ as

$$U^{(0)} = U(0), \qquad U^{(n)} = \lim_{s\to 0}(-1)^n \frac{d^n}{ds^n} U(s), \qquad n \geq 1.$$

Then, we obtain

$$U^{(0)} = I - [(-C)^{-1}D]^N, \qquad U^{(n)} = -[(-C)^{-1}D]^N S_N^{(n)}.$$

According to a similar reasoning in [Luca90], we obtain the following recursion to compute $T^{(n)}$:

$$\begin{aligned}
Z^{(n)} &= -nT^{(n-1)} + \sum_{k=0}^{n-1} \binom{n}{k} T^{(k)} S^{(n-k)} D - U^{(n)}, \qquad n \geq 1, \\
T^{(0)}\boldsymbol{e} &= \frac{1}{1-\rho}\left\{-U^{(1)}\boldsymbol{e} - E[S]\,U^{(0)}(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e}\right\}, \\
T^{(n)}\boldsymbol{e} &= \frac{E[S]}{1-\rho} Z^{(n)}(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e} + \frac{1}{(n+1)(1-\rho)} \\
&\qquad \cdot \left\{\sum_{k=0}^{n-1} \binom{n+1}{k} S^{(n+1-k)} T^{(k)} D\boldsymbol{e} - U^{(n+1)}\boldsymbol{e}\right\}, \qquad n \geq 1, \\
T^{(0)} &= T^{(0)}\boldsymbol{e}\boldsymbol{\pi} - U^{(0)}(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}, \\
T^{(n)} &= T^{(n)}\boldsymbol{e}\boldsymbol{\pi} + Z^{(n)}(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}, \qquad n \geq 1,
\end{aligned}$$

where $T^{(0)} = T(0)$. We summarize the procedure to compute $W^{(n)}$.

1. Compute $H_k^{(n)}$ and $S_k^{(n)}$ recursively.
2. Compute $T_k^{(n)}$ using $H_k^{(n)}$ and $S_k^{(n)}$.
3. Compute $U^{(0)}$, $T^{(0)}e$ and $T^{(0)}$ in this order.
4. Compute $U^{(n)}$, $Z^{(n)}$, $T^{(n)}e$ and $T^{(n)}$ recursively.
5. Compute $W^{(n)}$ using $T_k^{(n)}$ and $T^{(n)}$.

Remarks
1. Setting $N = 1$ in (6.25), we obtain

$$W^*(s) = \lambda^{-1} s \boldsymbol{y}_0 [sI + C + S^*(s)D]^{-1} De,$$

which is identical to the result in [Luca90].
2. In the case that messages arrive according to a Poisson process with rate λ, $C = -\lambda$ and $D = \lambda$. Substituting these into (6.10) yields $\boldsymbol{x}_0 = (1-\rho)/N$. Furthermore, (6.25) becomes

$$W^*(s) = \frac{1-\rho}{N} \frac{[\lambda/(s+\lambda)]^N - [S^*(s)]^N}{\lambda/(s+\lambda) - S^*(s)} + \frac{\lambda(1-\rho)\{1-[S^*(s)]^N\}}{N[s-\lambda+\lambda S^*(s)]},$$

which is the LST of the waiting time distribution of $M/G/1$ under N-policy [Taka91].

6.3 N-policy with Vacations

In this section, we consider a $MAP/G/1$ under N-policy with vacations in equilibrium. First, we consider the queue length distribution at departures. Then, we derive the formula of the queue length at an arbitrary time. We also derive the LST of the actual waiting time distribution for an arriving message.

6.3.1 Generating Function for Queue Length at Departures

We choose the time epochs immediately after the service termination and the vacation termination as imbedded points. Let $\boldsymbol{x}_n^{\mathrm{s}}$ ($\boldsymbol{x}_n^{\mathrm{v}}$) denote the joint probability vectors whose ith element represents the probability that the imbedded point is the service (vacation) termination, the number of the system is n and the phase of the arrival process is i. We define the following generating functions:

$$\boldsymbol{X}^{\mathrm{s}}(z) = \sum_{n=0}^{\infty} \boldsymbol{x}_n^{\mathrm{s}} z^n, \qquad \boldsymbol{X}^{\mathrm{v}}(z) = \sum_{n=0}^{\infty} \boldsymbol{x}_n^{\mathrm{v}} z^n, \qquad \boldsymbol{X}_k^{\mathrm{v}}(z) = \sum_{n=0}^{k} \boldsymbol{x}_n^{\mathrm{v}} z^n.$$

Let V_n denote an $m \times m$ vector whose (i,j)th element represents the conditional probability that n messages arrive during a vacation and the underlying Markov chain is in state j at the end of the vacation given that the underlying Markov chain being in state i at the beginning of the vacation. We define the matrix generating function $V(z)$ as

$$V(z) = \sum_{n=0}^{\infty} V_n z^n.$$

We then have [Luca90]

$$V(z) = \int_0^\infty e^{(C+zD)x}\, dV(x).$$

Considering the transition between consecutive imbedded points, we have the following equations:

$$\boldsymbol{X}^{\mathrm{s}}(z) = \left[\boldsymbol{X}^{\mathrm{s}}(z) - \boldsymbol{x}_0^{\mathrm{s}}\right] \frac{A(z)}{z} + \left[\boldsymbol{X}^{\mathrm{v}}(z) - \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)\right] \frac{A(z)}{z}, \tag{6.27}$$

$$\boldsymbol{X}^{\mathrm{v}}(z) = \boldsymbol{x}_0^{\mathrm{s}} V(z) + \boldsymbol{X}_{N-1}^{\mathrm{v}}(z) V(z), \tag{6.28}$$

$$\left[\boldsymbol{X}^{\mathrm{s}}(1) + \boldsymbol{X}^{\mathrm{v}}(1)\right] \boldsymbol{e} = 1, \tag{6.29}$$

where $A(z)$ is defined in (6.2).

Note that $\boldsymbol{x}_k^{\mathrm{v}}$ $(0 \le k \le N-1)$ are recursively obtained in terms of $\boldsymbol{x}_0^{\mathrm{s}}$:

$$\boldsymbol{x}_0^{\mathrm{v}} = \boldsymbol{x}_0^{\mathrm{s}} V_0 \left[I - V_0\right]^{-1}, \tag{6.30}$$

$$\boldsymbol{x}_k^{\mathrm{v}} = \left[\boldsymbol{x}_0^{\mathrm{s}} V_k + \sum_{i=0}^{k-1} \boldsymbol{x}_i^{\mathrm{v}} V_{k-i}\right] \left[I - V_0\right]^{-1}, \qquad 1 \le k \le N-1. \tag{6.31}$$

Thus, $\boldsymbol{X}^{\mathrm{v}}(z)$ is given in terms of $\boldsymbol{x}_0^{\mathrm{s}}$ (see (6.28)) and therefore $\boldsymbol{X}^{\mathrm{s}}(z)$ contains only one unknown vector $\boldsymbol{x}_0^{\mathrm{s}}$. Note that the queue length at departures is characterized by $\boldsymbol{X}^{\mathrm{s}}(z)$. Let $\boldsymbol{x}_k$ $(k \ge 0)$ denote a $1 \times m$ vector whose ith element represents the joint probability of k messages in the system and phase i of the underlying Markov chain at departures. Further, we define the vector generating function $\boldsymbol{X}(z)$ as

$$\boldsymbol{X}(z) = \sum_{k=0}^{\infty} \boldsymbol{x}_k z^k.$$

By definition, we have

$$\boldsymbol{X}(z) = \frac{\boldsymbol{X}^{\mathrm{s}}(z)}{\boldsymbol{X}^{\mathrm{s}}(1)\boldsymbol{e}}.$$

Thus once we obtain $\boldsymbol{x}_0^{\mathrm{s}}$, $\boldsymbol{X}(z)$ is completely determined. Before considering $\boldsymbol{x}_0^{\mathrm{s}}$, we derive some formulas which will be used later. Using (6.27), (6.28) and (6.29), we have the following equation:

$$\left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right) \boldsymbol{e} = \frac{1-\rho}{1-\rho+\lambda E[V]}. \tag{6.32}$$

The derivation of (6.32) is given in Appendix F. Using (6.32) and (F.3), we have

$$\boldsymbol{X}^{\mathrm{s}}(1)\boldsymbol{e} = \frac{\lambda E[V]}{1-\rho+\lambda E[V]},$$

and therefore we obtain

$$\boldsymbol{X}(z) = \frac{1-\rho+\lambda E[V]}{\lambda E[V]} \boldsymbol{X}^{\mathrm{s}}(z). \tag{6.33}$$

6.3.2 Computation of the Vector $\boldsymbol{x}_0^{\mathrm{s}}$

In this subsection, we derive a formula to compute $\boldsymbol{x}_0^{\mathrm{s}}$. First, we consider the number of messages at the end of an idle period when the threshold value is equal to n $(1 \le n \le N)$. Let R_k^{n} $(k \ge n)$ denote an $m \times m$ matrix whose (i, j)th element represents the conditional probability that there are k messages in the system and the underlying Markov chain is in state j at the end of an

idle period given that the underlying Markov chain being in state i at the beginning of the idle period. Note that the R_k^{n} is computed by the following recursion:

$$R_k^{\mathrm{1}} = [I - V_0]^{-1} V_k, \qquad k \geq 1, \tag{6.34}$$

$$R_k^{\mathrm{n}} = R_k^{\mathrm{n-1}} + R_{n-1}^{\mathrm{n-1}} \cdot R_{k-n+1}^{\mathrm{1}}, \qquad 2 \leq n \leq k. \tag{6.35}$$

For later use, we define the matrix generating function $R^{\mathrm{n}}(z)$ as

$$R^{\mathrm{n}}(z) = \sum_{k=n}^{\infty} R_k^{\mathrm{n}} z^k.$$

Now we consider the state transition during the recurrence time of the departure instant being in level zero. Let K denote an $m \times m$ matrix which represents the state transition matrix of the underlying Markov chain in the recurrence time. Furthermore, let $\boldsymbol{\kappa}$ denote the invariant probability vector of K. Then $\boldsymbol{x}_0^{\mathrm{s}}$ is given by

$$\boldsymbol{x}_0^{\mathrm{s}} = \frac{\boldsymbol{\kappa}}{\overline{K}},$$

where $\overline{K}$ denotes the mean recurrence time of the departure instant being in level zero.

Note that, with R_n^{N}, K is given by

$$K = \sum_{k=N}^{\infty} R_k^{\mathrm{N}} G^k,$$

where G is defined in (6.11). Thus, $\boldsymbol{\kappa}$ is obtained by solving $\boldsymbol{\kappa} K = \boldsymbol{\kappa}$ and $\boldsymbol{\kappa} \boldsymbol{e} = 1$. We now propose a simple recursive formula to compute $\overline{K}$. Multiplying both sides of (6.30) and (6.31) by $\overline{K}$, we obtain

$$\boldsymbol{x}_0^{\mathrm{v}*} = \boldsymbol{\kappa} V_0 [I - V_0]^{-1}, \tag{6.36}$$

$$\boldsymbol{x}_k^{\mathrm{v}*} = \left[\boldsymbol{\kappa} V_k + \sum_{i=0}^{k-1} \boldsymbol{x}_i^{\mathrm{v}*} V_{k-i} \right] [I - V_0]^{-1}, \qquad 1 \leq k \leq N-1, \tag{6.37}$$

where

$$\boldsymbol{x}_k^{\mathrm{v}*} = \overline{K} \boldsymbol{x}_k^{\mathrm{v}}.$$

Also, multiplying both sides of (6.32) by $\overline{K}$, we obtain

$$1 + \sum_{k=0}^{N-1} \boldsymbol{x}_k^{\mathrm{v}*} \boldsymbol{e} = \frac{1-\rho}{1-\rho+\lambda E[V]} \overline{K},$$

from which, it follows that

$$\overline{K} = \frac{1-\rho+\lambda E[V]}{1-\rho} \left(1 + \sum_{k=0}^{N-1} \boldsymbol{x}_k^{\mathrm{v}*} \boldsymbol{e} \right). \tag{6.38}$$

Therefore, $\overline{K}$ is computed as follows. First we compute $\boldsymbol{x}_k^{\mathrm{v}*}$ $(0 \leq k \leq N-1)$ by (6.36) and (6.37) and then compute $\overline{K}$ by (6.38).

6.3.3 Queue Length Distribution at Departures and its Moments

We first consider the queue length distribution at departures. Observing the system immediately after departures, we have the following transition matrix P:

$$P = \begin{bmatrix} O & O & O & \cdots & O & B_{N-1} & B_N & \cdots \\ A_0 & A_1 & A_2 & \cdots & A_{N-2} & A_{N-1} & A_N & \cdots \\ O & A_0 & A_1 & \cdots & A_{N-3} & A_{N-2} & A_{N-1} & \cdots \\ O & O & A_0 & \cdots & A_{N-4} & A_{N-3} & A_{N-2} & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \\ O & O & O & \cdots & A_0 & A_1 & A_2 & \cdots \\ O & O & O & \cdots & O & A_0 & A_1 & \cdots \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \end{bmatrix},$$

where

$$B_k = \sum_{n=N}^{k+1} R_n^{\mathrm{N}} A_{k+1-n}, \qquad k \geq N-1.$$

Since the transition matrix P takes the same form as in (6.1), we have the same recursion for $\boldsymbol{x}_k^{\mathrm{s}}$ as in section 6.2:

$$\boldsymbol{x}_k^{\mathrm{s}} = \left[\boldsymbol{x}_0^{\mathrm{s}} \overline{B}_k + \sum_{j=1}^{k-1} \boldsymbol{x}_j^{\mathrm{s}} \overline{A}_{k+1-j} \right] \left(I - \overline{A}_1 \right)^{-1},$$

where $\overline{A}_k$ and $\overline{B}_k$ are given in (6.14), (6.15) and (6.16). Thus, from (6.33), the queue length distribution $\boldsymbol{x}_k$ is computed by

$$\boldsymbol{x}_k = \frac{1-\rho+\lambda E[V]}{\lambda E[V]} \boldsymbol{x}_k^{\mathrm{s}}, \qquad k \geq 0.$$

Since the structure of the transition matrix is exactly the same as in section 6.2, we can use the same recursive formula in subsection 6.2.3 to compute the factorial moments for the queue length at departures.

6.3.4 Queue Length Distribution at an Arbitrary Time and its Moments

Let $\boldsymbol{Y}(z)$ denote the vector generating function of the number of messages at an arbitrary time. According to a similar reasoning as in subsection 6.2.4, we obtain

$$\begin{aligned} \boldsymbol{Y}(z) &= (1-\rho) \frac{\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)}{\left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right) e} V^*(z) \\ &\quad + \rho \frac{\boldsymbol{X}^{\mathrm{s}}(z) - \boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}^{\mathrm{v}}(z) - \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)}{1 - \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right) e} A^*(z), \end{aligned} \tag{6.39}$$

where $A^*(z)$ is given in (6.20) and $V^*(z)$ is the matrix generating function of the number of arrivals during the forward recurrence time of a vacation and given by:

$$V^*(z) = \frac{1}{E[V]} \left[V(z) - I \right] \left[C + zD \right]^{-1}.$$

Substituting $V^*(z)$ and $A^*(z)$ into (6.39) and noting the following equalities

$$\left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(z) \right] \left[V(z) - I \right] = \boldsymbol{x}_0^{\mathrm{s}} \left[R^{\mathrm{N}}(z) - I \right],$$

$$[\boldsymbol{X}^{\mathrm{s}}(z) - \boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}^{\mathrm{v}}(z) - \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)]\,[A(z) - I] = (z-1)\boldsymbol{X}^{\mathrm{s}}(z) - \boldsymbol{x}_0^{\mathrm{s}}\left[R^{\mathrm{N}}(z) - I\right],$$

we rewrite $\boldsymbol{Y}(z)$ as

$$\begin{aligned}
\boldsymbol{Y}(z) &= \frac{1-\rho}{E[V]}\frac{\boldsymbol{x}_0^{\mathrm{s}}}{(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1))\,\boldsymbol{e}}\left[R^{\mathrm{N}}(z) - I\right][C + zD]^{-1} \\
&\quad + \frac{\rho}{E[S]}\frac{(z-1)\boldsymbol{X}^{\mathrm{s}}(z) - \boldsymbol{x}_0^{\mathrm{s}}\left[R^{\mathrm{N}}(z) - I\right]}{1 - (\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1))\,\boldsymbol{e}}[C + zD]^{-1} \\
&= \frac{1 - \rho + \lambda E[V]}{E[V]}(z-1)\boldsymbol{X}^{\mathrm{s}}(z)\,[C + zD]^{-1} \\
&= \lambda(z-1)\boldsymbol{X}(z)\,[C + zD]^{-1}.
\end{aligned} \tag{6.40}$$

In (6.40), $\boldsymbol{X}(z)$ denotes the vector generating function of the queue length at departures (see (6.33)). Since this relationship holds for any stationary queue with MAP arrivals [Taki94a], an independent verification provides a validation for our analysis so far.

Since the queue length distributions at departures and at an arbitrary time are related by the common equation, the queue length distribution $\boldsymbol{y}_k$ at an arbitrary time is recursively obtained by (6.23) and (6.24) in terms of the queue length distribution $\boldsymbol{x}_k$ at departures. Furthermore using the recursion in section 6.2.4, we obtain the factorial moments of the queue length distribution at an arbitrary time.

6.3.5 Joint PDF of Number of Arrivals and Remaining Vacation Time

Let $\Omega(i,j,x)$ $(i,j = 0,1,\ldots)$ denote an $m \times m$ matrix whose (k,l)th element represents the probability that, given the phase being in i at the beginning of the vacation and a message arrival in the vacation, i messages arrive in the elapsed vacation time, j messages arrive in the remaining vacation time, the remaining vacation time is not greater than x and the phase is j at the end of the vacation. We also define the joint transformed matrix of $\Omega(i,j,x)$ as

$$\Omega^*(z_1,z_2,s) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\int_0^{\infty} z_1^i z_2^j e^{-sx}\,d\Omega(i,j,x).$$

Then, $\Omega^*(z_1,z_2,s)$ becomes

$$\begin{aligned}
\Omega^*(z_1,z_2,s) &= \int_0^{\infty}\frac{x\,dV(x)}{E[V]}\int_0^x \frac{dt}{x}e^{(C+z_1D)t}\cdot\frac{D}{\lambda}\cdot e^{(C+z_2D)(x-t)}e^{-s(x-t)} \\
&= \int_0^{\infty}\frac{dV(x)}{\lambda E[V]}\int_0^x dt\,e^{-\theta t}\sum_{m=0}^{\infty}\frac{t^m}{m!}(\theta I + C + z_1D)^m D e^{-\theta(x-t)} \\
&\quad \times\sum_{n=0}^{\infty}\frac{(x-t)^n}{n!}(\theta I + C + z_2D)^n e^{-s(x-t)} \\
&= \int_0^{\infty} e^{-(s+\theta)x}\frac{dV(x)}{\lambda E[V]}\int_0^x e^{st}dt\sum_{i=0}^{\infty}\sum_{n=0}^{i}\frac{t^{i-n}}{(i-n)!}\frac{(x-t)^n}{n!} \\
&\quad \times(\theta I + C + z_1D)^{i-n}D(\theta I + C + z_2D)^n.
\end{aligned} \tag{6.41}$$

In order to expand the matrix factor $(\theta I + C + z_1D)^k D(\theta I + C + z_2D)^l$, we introduce matrices $F_{k,l}(m,n)$ $(k,l = 0,1,2,\cdots,\ m = 0,1,\cdots,k,\ n = 0,1,\cdots,l)$ which satisfy

$$\sum_{m=0}^{k}\sum_{n=0}^{l} z_1^m z_2^n F_{k,l}(m,n) = (\theta I + C + z_1D)^k D(\theta I + C + z_2D)^l,$$

where $F_{0,0}(0,0) = D$. Then, matrices $F_{k,l}(m,n)$ satisfies the following recursion

$$F_{k+1,l}(m,n) = \begin{cases} (\theta I + C)F_{k,l}(0,n), & m = 0, \\ DF_{k,l}(m-1,n) + (\theta I + C)F_{k,l}(m,n), & 1 \le m \le k, \\ DF_{k,l}(k,n), & m = k+1, \end{cases} \tag{6.42}$$

and

$$F_{k,l+1}(m,n) = \begin{cases} F_{k,l}(m,0)(\theta I + C), & n = 0, \\ F_{k,l}(m,n-1)D + F_{k,l}(m,n)(\theta I + C), & 1 \le n \le l, \\ F_{k,l}(m,l)D, & n = l+1. \end{cases} \tag{6.43}$$

Thus, we obtain

$$\begin{aligned} &\sum_{i=0}^{\infty}\sum_{n=0}^{i} \frac{t^{i-n}}{(i-n)!}\frac{(x-t)^n}{n!}(\theta I + C + z_1 D)^{i-n} D (\theta I + C + z_2 D)^n \\ &= \sum_{i=0}^{\infty}\sum_{n=0}^{i} \frac{t^{i-n}}{(i-n)!}\frac{(x-t)^n}{n!} \sum_{l=0}^{i-n}\sum_{m=0}^{n} z_1^l z_2^m F_{i-n,n}(l,m) \\ &= \sum_{l=0}^{\infty}\sum_{m=0}^{\infty} z_1^l z_2^m \sum_{n=m}^{\infty}\sum_{i=l}^{\infty} \frac{t^i}{i!}\frac{(x-t)^n}{n!} F_{i,n}(l,m). \end{aligned} \tag{6.44}$$

Substituting (6.44) into (6.41) yields

$$\begin{aligned} \Omega^*(z_1,z_2,s) = &\sum_{l=0}^{\infty}\sum_{m=0}^{\infty} z_1^l z_2^m \\ &\times \left[\sum_{n=m}^{\infty}\sum_{i=l}^{\infty} \int_0^{\infty} e^{-(s+\theta)x}\frac{dV(x)}{\lambda E[V]}\int_0^x e^{st}dt\frac{t^i}{i!}\frac{(x-t)^n}{n!}F_{i,n}(l,m)\right]. \end{aligned} \tag{6.45}$$

Considering the coefficient matrices of $z_1^i z_2^j$ on both sides of (6.45), we obtain

$$\Omega(i,j,s) = \sum_{m=i}^{\infty}\sum_{n=j}^{\infty} \int_0^{\infty} e^{-(s+\theta)x}\frac{dV(x)}{\lambda E[V]}\int_0^x e^{st}dt\frac{t^m}{m!}\frac{(x-t)^n}{n!}F_{m,n}(i,j). \tag{6.46}$$

6.3.6 LST for Actual Waiting Time and its Moments

In this subsection, we consider the actual waiting time distribution for N-policy with vacations. Let $R_k^n(s)$ denote an $m \times m$ matrix whose (i,j)th element represents the LST for the length of the idle period when the number of messages is k and the phase is j at the end of the idle period given that the phase is i at the beginning of the idle period and the threshold value is n. From the similar reason of (6.34) and (6.35), $R_k^n(s)$ satisfies the following equations

$$R_k^1(s) = [I - V_0(s)]^{-1} V_k(s), \qquad k \ge 1, \tag{6.47}$$

$$R_k^n(s) = R_k^{n-1}(s) + R_{n-1}^{n-1}(s) \cdot R_{k-n+1}^1(s), \qquad 2 \le n \le k, \tag{6.48}$$

where

$$V_k(s) = \int_0^{\infty} e^{-st} P(k,t) dV(t).$$

We also define $R^{n}(s)$ as

$$R^{n}(s) = \sum_{k=n}^{\infty} R_k^{n}(s).$$

First, we consider the waiting time when the tagged message arrives at the system in a vacation time. We observe the the following two cases:

1. The queue length becomes greater than or equal to N at the end of the vacation time during which the tagged message arrives.
2. At the end of the vacation, there are k $(< N)$ messages in the system. Then, the next service starts after the period according to $R^{N-k}(s)$.

Thus, the LST $W_1^*(s)$ of the waiting time of a message when it arrives during a vacation time is given by

$$\begin{aligned} W_1^*(s) &= \frac{1-\rho}{(\boldsymbol{x}_0^{s} + \boldsymbol{X}_{N-1}^{v}(1))\,\boldsymbol{e}} \Bigg\{ (\boldsymbol{x}_0^{s} + \boldsymbol{x}_0^{v}) \sum_{i=0}^{\infty} \sum_{j=(N-i-1,0)^+}^{\infty} \Omega(i,j,s)[S^*(s)]^i \\ &\quad + \sum_{k=1}^{N-1} \boldsymbol{x}_k^{v} \sum_{i=0}^{\infty} \sum_{j=(N-k-i-1,0)^+}^{\infty} \Omega(i,j,s)[S^*(s)]^{k+i} \\ &\quad + (\boldsymbol{x}_0^{s} + \boldsymbol{x}_0^{v}) \sum_{i=0}^{N-2} \sum_{j=0}^{N-i-2} \Omega(i,j,s) R^{N-i-j-1}(s)[S^*(s)]^i \\ &\quad + \sum_{k=1}^{N-2} \boldsymbol{x}_k^{v} \sum_{i=0}^{N-k-2} \sum_{j=0}^{N-k-i-2} \Omega(i,j,s) R^{N-k-i-j-1}(s)[S^*(s)]^{k+i} \Bigg\} \boldsymbol{e}, \end{aligned} \tag{6.49}$$

where $(x,y)^+$ indicates the maximum value of x and y.

Next, we consider the waiting time when the server is busy. The joint transform $Y^*(z,s)$ defined in subsection 6.2.5 becomes

$$Y^*(z,s) = \rho \frac{\boldsymbol{X}^{s}(z) - \boldsymbol{x}_0^{s} + \boldsymbol{X}^{v}(z) - \boldsymbol{X}_{N-1}^{v}(z)}{1 - (\boldsymbol{x}_0^{s} + \boldsymbol{X}_{N-1}^{v}(1))\,\boldsymbol{e}} A(z,s).$$

Then, the LST $W_2^*(s)$ of the waiting time when the server is busy is given by

$$\begin{aligned} W_2^*(s) &= \frac{1}{1 - (\boldsymbol{x}_0^{s} + \boldsymbol{X}_{N-1}^{v}(1))\,\boldsymbol{e}} \\ &\quad \times [\boldsymbol{x}_0^{s} + \boldsymbol{X}_{N-1}^{v}(S^*(s))][I - V(S^*(s))][sI + C + S^*(s)D]^{-1} D\boldsymbol{e}. \end{aligned} \tag{6.50}$$

From (6.32), (6.49) and (6.50), the LST of the actual waiting time distribution is obtained as

$$\begin{aligned} W^*(s) &= W_1^*(s) + W_2^*(s) \\ &= \frac{1-\rho+\lambda E[V]}{\lambda E[V]} \Bigg[\lambda E[V] \Bigg\{ (\boldsymbol{x}_0^{s} + \boldsymbol{x}_0^{v}) \sum_{i=0}^{\infty} \sum_{j=(N-i-1,0)^+}^{\infty} \Omega(i,j,s)[S^*(s)]^i \\ &\quad + \sum_{k=1}^{N-1} \boldsymbol{x}_k^{v} \sum_{i=0}^{\infty} \sum_{j=(N-k-i-1,0)^+}^{\infty} \Omega(i,j,s)[S^*(s)]^{k+i} \end{aligned}$$

$$+(\boldsymbol{x}_0^{\mathrm{s}}+\boldsymbol{x}_0^{\mathrm{v}})\sum_{i=0}^{N-2}\sum_{j=0}^{N-i-2}\Omega(i,j,s)R^{\mathrm{N-i-j-1}}(s)[S^*(s)]^i$$
$$+\sum_{k=1}^{N-2}\boldsymbol{x}_k^{\mathrm{v}}\sum_{i=0}^{N-k-2}\sum_{j=0}^{N-k-i-2}\Omega(i,j,s)R^{\mathrm{N-k-i-j-1}}(s)[S^*(s)]^{k+i}\Bigg\}$$
$$+[\boldsymbol{x}_0^{\mathrm{s}}+\boldsymbol{X}_{N-1}^{\mathrm{v}}(S^*(s))][I-V(S^*(s))][sI+C+S^*(s)D]^{-1}D\Bigg]\boldsymbol{e}. \qquad (6.51)$$

For calculating nth moment of the waiting time, we define following notations:

$$T_{ijk}(s)=\Omega(i,j,s)\,[S^*(s)]^{k+i},\qquad T_{ijk}^{(n)}=\lim_{s\to 0}(-1)^n\frac{d^n}{ds^n}T_{ijk}(s),$$

$$U_{ijk}(s)=\Omega(i,j,s)\,R^{\mathrm{N-k-i-j-1}}(s)\,[S^*(s)]^{k+i},\qquad U_{ijk}^{(n)}=\lim_{s\to 0}(-1)^n\frac{d^n}{ds^n}U_{ijk}(s),$$

$$\boldsymbol{x}(s)=\boldsymbol{x}_0^{\mathrm{s}}+\boldsymbol{X}_{N-1}^{\mathrm{v}}(S^*(s)),\qquad \boldsymbol{x}^{(n)}=\lim_{s\to 0}(-1)^n\frac{d^n}{ds^n}\boldsymbol{x}(s),$$

$$T(s)=U(s)\,[sI+C+S^*(s)D]^{-1},\qquad U(s)=I-V(S^*(s)).$$

$T^{(n)}$ and $U^{(n)}$ are defined in subsection 6.2.5. Then, the nth moment $W^{(n)}$ of the actual waiting time becomes

$$W^{(n)} = \frac{1-\rho+\lambda E[V]}{\lambda E[V]}\Bigg[\lambda E[V]\Bigg\{(\boldsymbol{x}_0^{\mathrm{s}}+\boldsymbol{x}_0^{\mathrm{v}})\sum_{i=0}^{\infty}\sum_{j=(N-i-1,0)^+}^{\infty}T_{ij0}^{(n)}$$
$$+\sum_{k=1}^{N-1}\boldsymbol{x}_k^{\mathrm{v}}\sum_{i=0}^{\infty}\sum_{j=(N-k-i-1,0)^+}^{\infty}T_{ijk}^{(n)}+(\boldsymbol{x}_0^{\mathrm{s}}+\boldsymbol{x}_0^{\mathrm{v}})\sum_{i=0}^{N-2}\sum_{j=0}^{N-i-2}U_{ij0}^{(n)}$$
$$+\sum_{k=1}^{N-2}\boldsymbol{x}_k^{\mathrm{v}}\sum_{i=0}^{N-k-2}\sum_{j=0}^{N-k-i-2}U_{ijk}^{(n)}\Bigg\}+\sum_{m=0}^{n}\binom{n}{m}\boldsymbol{x}^{(m)}T^{(n-m)}D\Bigg]\boldsymbol{e}.$$

From definitions of $T_{ijk}(s)$ and $U_{ijk}(s)$, $T_{ijk}^{(n)}$ and $U_{ijk}^{(n)}$ becomes

$$T_{ijk}^{(n)} = \sum_{m=0}^{n}\binom{n}{m}\Omega^{(m)}(i,j)\,S_{k+i}^{(n-m)},$$
$$U_{ijk}^{(n)} = \sum_{m=0}^{n}\binom{n}{m}\Bigg\{\sum_{l=0}^{m}\binom{m}{l}\Omega^{(l)}(i,j)\,R^{\mathrm{N-k-i-j-1(m-l)}}\Bigg\}S_{k+i}^{(n-m)},$$

where

$$\Omega^{(n)}(i,j)=\lim_{s\to 0}(-1)^n\frac{d^n}{ds^n}\Omega(i,j,s),\qquad R^{\mathrm{m}(n)}=\lim_{s\to 0}(-1)^n\frac{d^n}{ds^n}R^{\mathrm{m}}(s).$$

From (6.46), we obtain

$$\Omega^{(k)}(i,j) = \frac{1}{\lambda E[V]}\sum_{m=i}^{\infty}\sum_{n=j}^{\infty}\sum_{l=0}^{n+k}\frac{F_{m,n}(i,j)}{m!n!}\frac{(-1)^l}{l+m+1}\binom{n+k}{l}$$
$$\cdot\int_0^{\infty}x^{m+n+k+1}e^{-\theta x}dV(x).$$

We define $R_k^{\mathrm{m}(n)}$ as

$$R_k^{\mathrm{m}(n)} = \lim_{s\to 0}(-1)^n \frac{d^n}{ds^n} R_k^{\mathrm{m}}(s).$$

Using $R_k^{\mathrm{m}(n)}$, $R^{\mathrm{m}(n)}$ is expressed as

$$R^{\mathrm{m}(n)} = \sum_{k=m}^{\infty} R_k^{\mathrm{m}(n)}.$$

From (6.47) and (6.48), $R_k^{\mathrm{m}(n)}$ can be expressed as

$$R_k^{1(0)} = [I - V_0]^{-1} V_k, \qquad R_k^{1(n)} = [I - V_0]^{-1} \left[V_k^{(n)} + \sum_{l=0}^{n-1} \binom{n}{l} V_0^{(n-l)} R_k^{1(l)} \right],$$

$$R_k^{\mathrm{m}(n)} = R_k^{\mathrm{m}-1(n)} + \sum_{l=0}^{n} \binom{n}{l} R_{m-1}^{\mathrm{m}-1(l)} R_{k-m+1}^{1(n-l)}, \qquad 2 \le m \le k.$$

Hence, we can calculate $R_k^{\mathrm{m}(n)}$ recursively.

$x^{(n)}$ can be calculated from the following equations:

$$x^{(n)} = \begin{cases} x_0^{\mathrm{s}} + X_{N-1}^{\mathrm{v}}(1), & n = 0, \\ \sum_{k=1}^{N-1} x_k^{\mathrm{v}} S_k^{(n)}, & n > 0. \end{cases}$$

Since we can calculate $T^{(n)}$ according to the same way of the N-policy without vacations, we consider the calculating formula of $U^{(n)}$. According to [Taki93b], $V(z)$ can be rewritten as

$$V(z) = \sum_{m=0}^{\infty} \zeta_m [I + \theta^{-1}(C + zD)]^m = \sum_{k=0}^{\infty} z^k \sum_{m=k}^{\infty} \zeta_m F_m(k),$$

where

$$\zeta_m = \int_0^{\infty} \frac{(\theta x)^m}{m!} e^{-\theta x} dV(x),$$

and $F_m(k)$ satisfies following equations:

$$F_{m+1}(k) = \begin{cases} (I + \theta^{-1}C)^{m+1}, & k = 0, \\ F_m(k)\,(I + \theta^{-1}C) + F_m(k-1)\,(\theta^{-1}D), & 1 \le k \le m, \\ (\theta^{-1}D)^{m+1}, & k = m+1. \end{cases}$$

Then, $U^{(n)}$ can be calculated from following equations:

$$U^{(n)} = \begin{cases} I - V, & n = 0, \\ -\sum_{m=0}^{\infty} \zeta_m \sum_{k=0}^{m} F_m(k)\, S_k^{(n)}, & n > 0. \end{cases}$$

We summarize the procedure to compute $W^{(n)}$.

1. Compute $R_k^{N(n)}$ and then $R^{\mathrm{N}(n)}$.

2. Compute $F_{k,l}(i,j)$ and then $\Omega^{(n)}(i,j)$.

3. Compute $T_{ijk}^{(n)}$ and $U_{ijk}^{(n)}$ using $\Omega^{(n)}(i,j)$, $R^{N-k-i-j-1(n)}$ and $S_{k+i}^{(n)}$.

4. Compute $U^{(n)}$ and then $T^{(n)}$ in the similar manner of section 6.2.5.

5. Compute $x^{(n)}$.

6. Finally, compute $W^{(n)}$ using $T_{ijk}^{(n)}$, $U_{ijk}^{(n)}$, $x^{(n)}$ and $T^{(n)}$.

6.4 Numerical Examples

In this section, we present some numerical examples of the mean waiting times for N-policy with and without vacations. In our numerical examples, the service time distribution is chosen as an unit distribution with mean $E[S] = 1.0$ and the vacation time distribution as an exponential distribution with mean $E[V] = 1.0$. The arrival process is assumed to be a 2-state $MMPP$ with

$$C = \begin{pmatrix} -r - \frac{2}{11}\rho & r \\ r & -r - \frac{20}{11}\rho \end{pmatrix}, \qquad D = \begin{pmatrix} \frac{2}{11}\rho & 0 \\ 0 & \frac{20}{11}\rho \end{pmatrix}.$$

From this construction, it is easy to see that $\pi = (1/2, 1/2)$. Note that the correlation in the arrival process becomes large with the decrease of r. We calculate the mean waiting times with $r = 0.5$ and 1.0.

In computing the matrices G and A under both N-policy with and without vacations, we truncate the infinite sums according to the criteria proposed in [Taki93b]. We also truncate the infinite sum for calculating V using the same criterion.

In computing the kth moment $\Omega^{(k)}(i,j)$, we need to truncate the infinite sums of (6.46). The accuracy of $\Omega^{(k)}(i,j)$ depends on how many number of arrays for $F_{k,l}(m,n)$ we can store. Let c denote the index of the set $\{F_{k,l}(m,n) : 0 \le m \le k,\ 0 \le n \le l\}$, where $c = k + l$. From (6.42) and (6.43), the $c+1$st set of $F_{k,l}(m,n)$ can be calculated using the cth set of $F_{k,l}(m,n)$ (see Fig.6.1). Note that we choose a maximum value $c_{\max}$ of c under the constraint of computer resources such as disk space and memory size. In our implementation, we set $c_{\max}$ to be 34. Since $\pi\Omega^*(1,1,0)e = 1$, we can check the accuracy of $\Omega^{(k)}(i,j)$ by summing $\Omega^{(0)}(i,j)$ over all i and j we computed.

We first compare the mean waiting times calculated from moment formulas with those calculated from Little's formula using the mean queue length $Y^{(1)}$. Tables 6.1 and 6.2 show the numerical results of N-policy without and with vacations, respectively, where $N = 5$ and $r = 1.0$. In those tables, W_{LST} denotes the mean waiting time calculated by the LST and W_{Little} denotes that by Little's formula.

From Table 6.1, we observe that W_{LST} gives good agreement with W_{Little}. On the other hand, Table 6.2 shows that $|W_{LST} - W_{Little}|$ increases as ρ becomes large. This is because the accuracy of $\pi\Omega^*(1,1,0)e$ becomes worse (recall that we fixed $c_{\max}$ to 34). Note, however, that W_{LST} agrees with W_{Little} in the order of 10^{-4} as $\rho = 0.9$. Thus it seems that it is sufficient for graphic representations to set $c_{\max} = 34$, except for the region of very high traffic. Therefore we use the result calculated by the LST in the following figures.

Fig.6.2 shows the mean waiting times in the case of $N = 5$ and 10 with $r = 1.0$. We observe that the mean waiting time becomes large as the value of N increases, and that the mean waiting time under N-policy with vacations is always larger than that without vacations. We

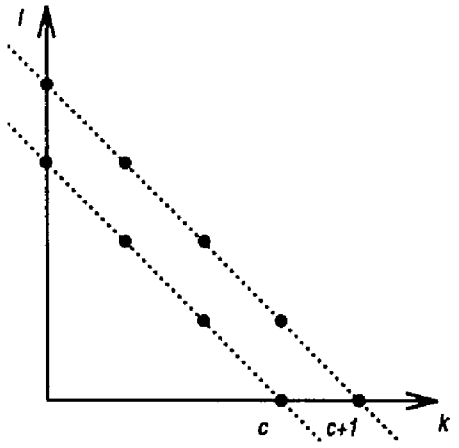


Figure 6.1: cth and $c+1$st sets of $F_{k,l}(m,n)$

ρ	W_{LST}	W_{Little}	$\lvert W_{LST} - W_{Little}\rvert$
0.10000	19.9516	19.9516	0.00000
0.20000	10.0602	10.0602	0.00000
0.30000	6.86860	6.86860	0.00000
0.40000	5.39333	5.39333	0.00000
0.50000	4.66432	4.66432	0.00000
0.60000	4.40793	4.40793	0.00000
0.70000	4.62049	4.62049	0.00000
0.80000	5.64797	5.64797	0.00000
0.90000	9.53749	9.53749	0.00000

Table 6.1: Comparison of Mean Waiting Times under N-policy without Vacations.

also observe that mean waiting times in all cases diverge to infinity as ρ becomes small. This is because the queue length is hard to reach N when ρ is small.

To investigate the influence of the correlation in arrivals on the mean waiting time, we plot Figs.6.3 and 6.4, which show the mean waiting times with $r = 0.5, 1.0$ and that in Poisson arrivals with the same arrival rate, where $N = 5$. We observe that when ρ is large, the mean waiting time becomes large with the increase of the correlation in arrivals (recall that the correlation in arrivals becomes high with the decrease of r). However, when ρ is small, higher correlation leads to a smaller value of the mean waiting time. Please also see Table 6.3, which give numerical data of Figs.6.3 and 6.4, respectively.

From these tables, we observe that when ρ is small,

$$W_{r=0.5} < W_{r=1.0} < W_{Poisson},$$

and when ρ is large,

$$W_{r=0.5} > W_{r=1.0} > W_{Poisson}.$$

In general, higher correlation in arrival makes the mean waiting time larger. However, our numerical results show that it is not the case. Note that, in N-policy, the mean waiting time

ρ	W_{LST}	W_{Little}	$\lvert W_{LST} - W_{Little}\rvert$	$\pi\Omega^*(1,1,0)e$
0.10000	20.4579	20.4579	0.00000	0.99999
0.20000	10.6128	10.6128	0.00000	0.99999
0.30000	7.47885	7.47885	0.00000	0.99999
0.40000	6.07984	6.07984	0.00000	0.99999
0.50000	5.45828	5.45828	0.00000	0.99999
0.60000	5.36508	5.36508	0.00000	0.99998
0.70000	5.85289	5.85288	0.00001	0.99995
0.80000	7.43537	7.43533	0.00003	0.99991
0.90000	12.9965	12.9965	0.00003	0.99983

Table 6.2: Comparison of Mean Waiting Times under N-policy with Vacations

	N-policy without vacations			N-policy with vacations		
ρ	$r = 0.5$	$r = 1.0$	$W_{Poisson}$	$r = 0.5$	$r = 1.0$	$W_{Poisson}$
0.01000	199.74357	199.87384	200.00505	200.34455	200.47517	200.60585
0.02000	99.75333	99.88173	100.01020	100.35618	100.48438	100.61180
0.03000	66.42999	66.55646	66.68213	67.03470	67.16042	67.28452
⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.36000	5.85972	5.86487	5.83681	6.79160	6.79476	6.76540
0.37000	5.73609	5.73469	5.69906	6.64972	6.64650	6.60968
0.38000	5.62130	5.61307	5.56961	6.51801	6.50818	6.46367

Table 6.3: Numerical Results under N-policy with and without Vacations

$E[W]$ consists of two terms; one is the mean waiting time $E[W_1]$ of messages which arrive in the idle period and the other is the mean waiting time $E[W_2]$ of messages which arrive in the busy period. Namely,

$$E[W] = (1 - \rho)E[W_1] + \rho E[W_2]$$

Tables 5 and 6 show $E[W_1]$ and $E[W_2]$ in the same settings as in Tables 3 and 4. We observe that $E[W_1]$ (resp. $E[W_2]$) is a decreasing (resp. an increasing) function of correlation in arrivals for a fixed ρ. We explain this phenomenon. When the correlation in arrivals is high, messages arrive back to back once a message arrives. Thus after the first message arrives in the idle period, subsequent messages are likely to arrive in a short interval, so that the mean waiting time of those messages becomes small according to the increase of the correlation in arrivals. On the other hand, the mean waiting time of messages which arrive in the busy period becomes large with the increase of correlation in arrivals, as in a work-conserving queue. In light traffic (i.e., for a small ρ), the former is the dominant factor in the mean waiting time $E[W]$. Thus, correlation in arrivals leads to a smaller mean waiting time in light traffic.

	$E[W_1]$			$E[W_2]$		
ρ	$r = 0.5$	$r = 1.0$	$W_{Poisson}$	$r = 0.5$	$r = 1.0$	$W_{Poisson}$
0.01000	201.73042	201.86440	202.00000	3.04533	2.80832	2.50505
0.02000	101.72677	101.86267	102.00000	3.05443	2.81556	2.51020
0.03000	68.38977	68.52759	68.66667	3.06376	2.82296	2.51546
⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.36000	7.14511	7.35454	7.55556	3.57458	3.21655	2.78125
0.37000	6.99066	7.20240	7.40541	3.59992	3.23560	2.79365
0.38000	6.84411	7.05815	7.26316	3.62618	3.25531	2.80645

Table 6.4: Numerical Results under N-policy without Vacations

	$E[W_1]$			$E[W_2]$		
ρ	$r = 0.5$	$r = 1.0$	$W_{Poisson}$	$r = 0.5$	$r = 1.0$	$W_{Poisson}$
0.01000	202.34008	202.47375	202.60681	2.78663	2.61574	2.51106
0.02000	102.34701	102.48138	102.61363	2.80528	2.63125	2.52224
0.03000	69.02059	69.15569	69.28712	2.82417	2.64696	2.53354
⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.36000	8.12492	8.29845	8.40809	3.66128	3.32548	3.00692
0.37000	7.98094	8.15603	8.26516	3.69670	3.35345	3.02585
0.38000	7.84483	8.02151	8.13015	3.73306	3.38209	3.04519

Table 6.5: Numerical Results under N-policy with Vacations

6.5 Conclusion

In this chapter, we have considered queueing systems under N-policy with and without vacations. In both models, we have obtained the queue length distribution at departure epochs, that at an arbitrary time and the LST of the actual waiting time distribution. We also showed the numerical examples of the mean waiting times of both models. From numerical examples, we have shown that in light traffic, the correlation in arrivals leads to a smaller mean waiting time.

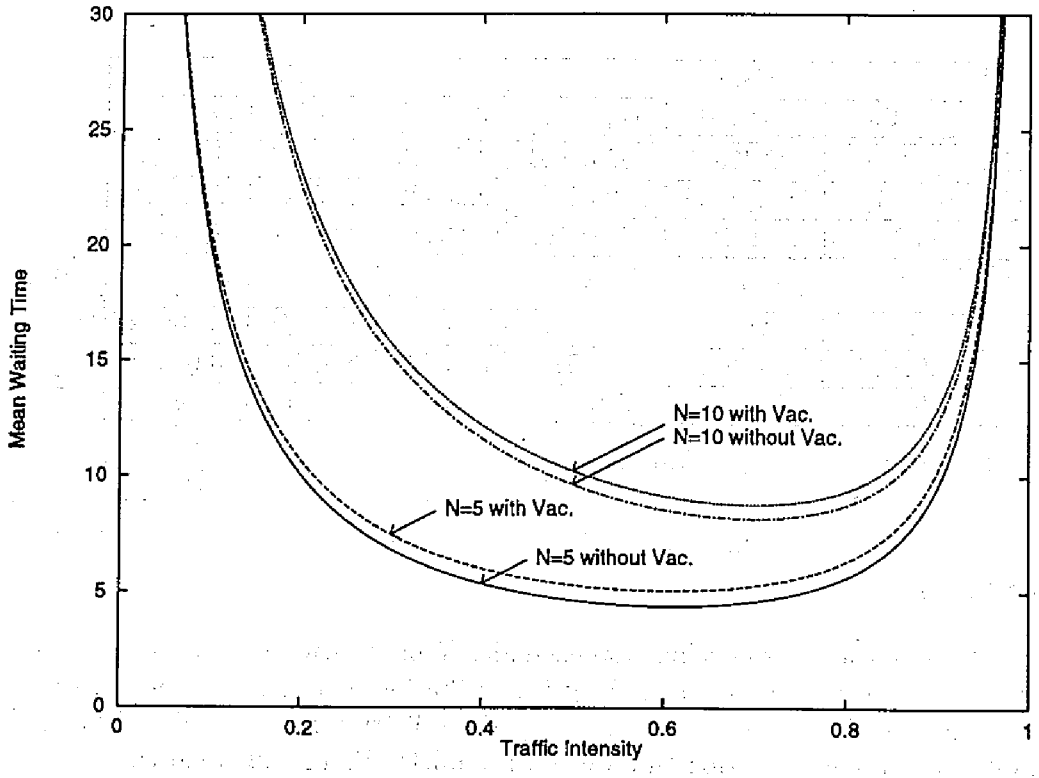


Figure 6.2: Mean Waiting Times under N-policy with and without Vacations

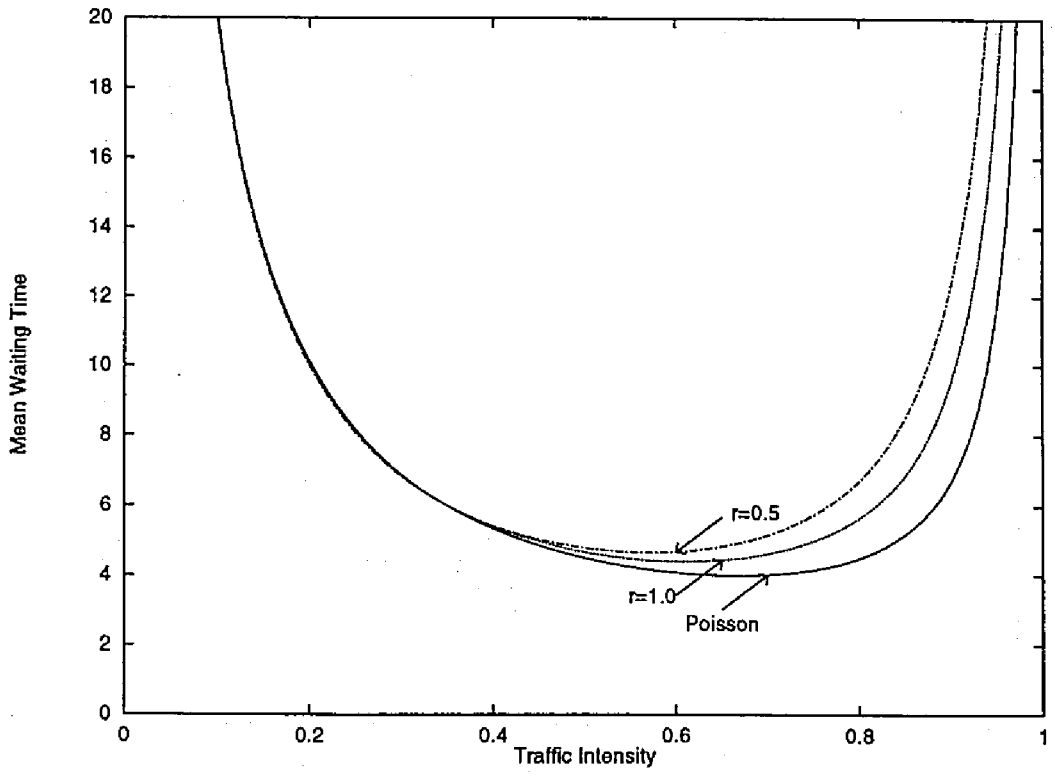


Figure 6.3: Mean Waiting Time under N-policy without Vacations

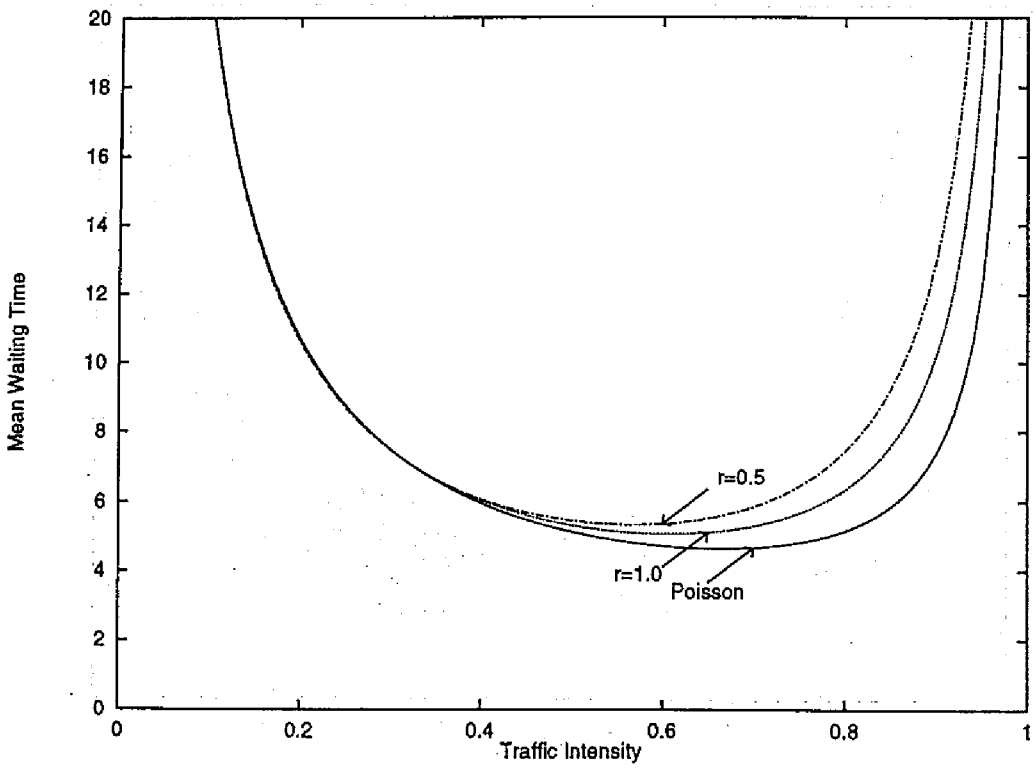


Figure 6.4: Mean Waiting Time under N-policy with Vacations

Chapter 7

Concluding Remarks

In this dissertation, we have considered the queueing systems with vacations concerning a finite buffer, buffer control policies and non-Poissonian arrival processes. We summarize the results of this dissertation in section 7.1 and finally present some topics for future researches in section 7.2.

7.1 Summary of Results

The results obtained in this dissertation are described as below:

In Chapter 2, we considered an $M/G/1/K$ system without vacations under random scheduling and LCFS. The LSTs of the waiting time distributions for both disciplines were derived and the mean and the coefficient of variation were calculated under several conditions. From the numerical results, it turned out that mean waiting times under three service disciplines are the same and approach $(K-1)b$ when the offered load becomes large. In addition, the c.v. of FCFS is smallest and that of LCFS largest. It means that variations of the waiting time under FCFS, random scheduling and LCFS become large in this order.

In Chapter 3, an $M/G/1/K$ system with multiple vacations and the exhaustive service discipline were studied. Similarly to Chapter 2, the LSTs of waiting time distributions under random scheduling and LCFS were derived. We also presented the numerical algorithms for the moments of the waiting time and then discussed the numerical results. In addition to the similar results of Chapter 2, it turned out that the waiting time is influenced by vacations under light offered load and that the waiting time approaches the remaining vacation time when the offered load becomes small. From the numerical examples of the c.v.'s of the waiting time compared among three service disciplines, we discussed the limiting behavior of the c.v.'s of FCFS and random scheduling, respectively.

In Chapter 4, we studied an $M/G/1/K$ system with push-out scheme and multiple vacations under FCFS and LCFS. We derived the LST of the waiting time distribution for messages which are eventually served, and the mean waiting time for pushed-out messages. Using these results, we calculated the mean waiting times and c.v.'s under several situations. From the numerical examples, it turned out that the mean waiting times of PB-served and PB-pushed out messages converge as the offered load becomes large, and that those limiting values are smaller than that under NPB case. We also observed that the variation of the waiting time of the PB-served message is small and stable in comparison with that of the NPB one.

In Chapter 5, we considered an $SPP/G/1$ queue with multiple vacations and E-limited discipline. Using the supplementary variable technique, we obtained the transform of the stationary queue length distribution explicitly. From the numerical examples, we observed that the mean waiting time becomes quite large around the upper boundary of the arrival rate, which is determined by the equilibrium condition. The mean waiting time is also affected by the ratio of two arrival rates even when the overall arrival rate is fixed. Furthermore, the mean waiting time is affected strongly by the arrival rate and the mean sojourn time in each state of the arrival process.

In Chapter 6, we studied $MAP/G/1$ queues under N-policy with and without vacations. For each case, we analyzed the stationary queue length and the actual waiting time distributions, and derived the recursive formulas to compute the moments of these distributions. In numerical examples, we observed that the mean waiting time becomes large as the value of N increases, and that the mean waiting time under N-policy with vacations is always larger than that without vacations. We also observed that when the offered load is large, the mean waiting time becomes large with the increase of the correlation in arrivals. However, when the offered load is small, higher correlation leads to a smaller value of the mean waiting time. This implies that the mean waiting time is significantly influenced by N-policy under the light traffic.

7.2 Future Research Topics

Queueing systems with vacations have been studied extensively in last two decade. Recently, queueing systems with a non-Markovian arrival process and a generalized service time process like a semi-Markov process become more popular than ever in this field, but there are still open problems concerning vacations and service disciplines. The author thinks that the following topics are worth to analyze:

- We can extend the models treated in Chapter 6 to $BMAP/G/1$ queues. In this model, the performance measures should be studied considering the influence of $BMAP$ arrivals.
- There are few studies in $MAP/G/1$ queues with a finite buffer under buffer control policies like PB. It is significant to analyze those models and to investigate the influence of buffer control policies.
- Concerning the $GI/M/1$-type, queueing systems whose successive service times form a semi-Markov (SM) process have been studied. In particular, $GI/SM/1$, $SM/SM/1$, and $MAP/SM/1$ have been analyzed in [Seng89, Seng90]. In those models, the queue length and the waiting time have been mainly analyzed but service disciplines have not been considered in detail. It is worthwhile to study those models with and without vacations under several service disciplines.
- The queueing system with vacations in which the successive vacation time forms a semi-Markov process is valuable to analyze. In particular, it is significant to analyze the $MAP/G/1$ with semi-Markovian vacation process.
- In general, the matrix analytical approach used in Chapter 6 needs the enormous resources of the computer systems such as memory and hard disk since there are a number of states to be considered. Hence, the effective method to reduce the number of states is an important problem for the implementation of the numerical algorithms.

Appendix A

Glossary of Principal Symbols

The following is a list of the principal symbols that appear in this dissertation. A brief description of each symbol is also given. Page numbers indicate where these symbols are defined for the first time.

Symbol	*Definition*	*Page*
A	Number of the arrivals during the remaining service time	21
A	Abbreviation of $A(1)$	92
A_n	$m \times m$ matrix that characterizes the transition probability matrix for $MAP/G/1$	91
$A(z)$	Matrix generating function of A_n	91
$A(z,s)$	Joint transformed matrix for the number of messages which arrive in the backward recurrence time and the forward recurrence time	97
$A^*(z)$	Matrix generating function of the number of arrivals during the remaining service time	95
$A^{(n)}$	nth factorial moment of $A(z)$	94
B_n	$m \times m$ matrix that characterizes the transition probability matrix for $MAP/G/1$	91
$B(z)$	Matrix generating function of B_n	91
$B^{(n)}$	nth factorial moment of $B(z)$	94
C	Stable matrix that characterizes the MAP	8
C_T	c.v. of the sojourn time in the system	22
C_W	c.v. of the waiting time	22
D	Non-negative matrix that characterizes the MAP	8
$E[X]$	Expectation of a random variable	38
G	State transition matrix of the underlying Markov chain during the first passage time	93
I, $\boldsymbol{I}$	Identity matrix	8, 80
$J(t)$	State of the Markov process at time t	8
K	System capacity including the server	16
K	State transition matrix of the underlying Markov chain during the recurrence time	93
$\overline{K}$	Mean recurrence time of level zero	93
L	Number of messages in the system at arbitrary instant	18

Symbol	*Definition*	*Page*
L_n	Number of messages in the system immediately after the nth Markov point	16
$L(z)$	Generating function for the queue length at arbitrary instant	84
$\overline{L}$	Mean queue length	84
M	Pre-specified value for E–limited service discipline	76
N	Threshold value of messages in the buffer for N-policy	2
$N(t)$	Number of arrivals in $(0, t]$	8
P	Transition probability matrix for $MAP/G/1$	91
$P(n, t)$	Transition matrix of the number of arrivals and the state of the Markov process at timet	8
$P^*(z, t)$	Matrix generating function of $P(n, t)$	9
P_k	Probability distribution of the number of messages at arbitrary instant	17
$P_m^{(l)}(z, x)$	Generating function of $P_{k,m}^{(l)}(x)$	78
$P_m^{*(l)}(z, s)$	Joint transformed function for the number of messages and the attained service time	82
P_B	Loss probability	16
$P_{k,m}^{(l)}(\cdot)$	Joint PDF of number of messages, server state, phase of the arrival process and attained service time	76
Q	Infinitesimal generator of the m-state continuous-time Markov chain	9
$Q^{*(l)}(z, s)$	Joint transformed function for the number of messages and the attained vacation time	82
$Q_k^{(l)}(\cdot)$	Joint PDF of number of messages, server state, phase of the arrival process and attained vacation time	76
R_k^n	Conditional probability matrix for the number of messages at the beginning of the service period under n-policy with vacations	100
$R^n(z)$	Matrix generating function of R_k^n	101
S	Service time	76
$S(\cdot)$	Service time distribution	16
$\widehat{S}$	Attained service time	18
$\widetilde{S}$	Remaining service time	18
$\widetilde{S}(\cdot)$	Attained service time distribution	63
$\widetilde{S}_j^*(s)$	LST for attained service time and the number of arrivals	63
$S_k^*(s)$	LST for the service time and the number of arrivals	20
$T^*(s)$	LST for the sojourn time	22
U	Idle time of the server	95
V	Vacation time	76
$V(\cdot)$	Vacation time distribution	37
V_n	$m \times m$ matrix that characterizes the transition probability matrix for $MAP/G/1$ with vacations	99
$V(z)$	Matrix generating function of V_n	99
$\widetilde{V}$	Remaining vacation time	39
$\widetilde{V}(\cdot)$	Attained vacation time distribution	63
$\widetilde{V}^{(n)}$	nth moment of the remaining vacation time	45

Symbol	*Definition*	*Page*
$\tilde{V}_j^*(s)$	LST for attained vacation time and the number of arrivals	63
$V^*(z)$	Matrix generating function of the number of arrivals during the forward recurrence time of a vacation	102
$V_k(s)$	LST for the vacation time and the number of *MAP* arrivals	105
$\bar{W}$	Mean waiting time	22
$\widetilde{W}_k^*(s)$	LST for $\widetilde{W}_k$	63
$\widetilde{W}_k$	Waiting time of a tagged message that has k other messages ahead at the end of a service or a vacation	63
$W^{(n)}$	nth moment of the actual waiting time	97
$W^{(n)}$	nth moment of the waiting time	22
$W^*(s)$	LST for the waiting time	20
$W_1^*(s)$	LST for waiting time when the message arrives during an idle time of the server	96
$W_2^*(s)$	LST for the waiting time of a message which arrives when the server is busy	97
W_B	Waiting time for NPB model	61
W_P	Waiting time for PB model	61
$W_j(x)$	Probability that the service of an arbitrary message among the j messages in the system starts within time x from an imbedded point	20
$W_j^*(s)$	LST of $W_j(x)$	20
$W_{k:n}$	Waiting time of a message that has k other messages ahead and n others behind it at the end of a service or a vacation	62
$W_{k:n}^*(s)$	LST for $W_{k:n}$	62
$\boldsymbol{X}_k^{\mathrm{v}}(z)$	Vector function for $\boldsymbol{x}_n^{\mathrm{v}}$	99
$\boldsymbol{X}^{\mathrm{s}}(z)$	Vector generating function of $\boldsymbol{x}_n^{\mathrm{s}}$	99
$\boldsymbol{X}^{\mathrm{v}}(z)$	Vector generating function of $\boldsymbol{x}_n^{\mathrm{v}}$	99
$\boldsymbol{X}(z)$	Vector generating function of $\boldsymbol{x}_k$	91
$\boldsymbol{X}^{(n)}$	nth factorial moment of $\boldsymbol{X}(z)$	94
$\boldsymbol{Y}(z)$	Vector generating function of $\boldsymbol{y}_k$	95
$Y^*(z,s)$	Joint transform of the number of messages and the forward recurrence time at an arbitrary point of the current service	96
Λ	Diagonal matrix whose (i,i)th element is λ_i	9
$\Omega(i,j,x)$	Conditional probability matrix for the remaining vacation time and the number of arrivals in the elapsed, and vacation times	103
$\Omega^*(z_1,z_2,s)$	Transformed matrix of $\Omega(i,j,x)$	103
$\Omega_{j:k}^*(s)$	LST for the number of messages, arrivals and remaining vacation time	41
$\Omega_k(x)$	Inverse transforms of $\Omega_k^*(s)$	39
$\Omega_k^*(s)$	LST for the number of messages and the remaining vacation time	39
$\Pi_{j:k}^*(s)$	LST for the number of messages, arrivals and remaining service time	21
Π_k	Probability that an arriving message finds k messages in the system	17
$\Pi_j(x)$	Inverse transform of $\Pi_j^*(s)$	21
$\Pi_k^*(s)$	LST for number of messages and remaining service time	18

Symbol	*Definition*	*Page*
α	Rate of the underlying Markov process for *SPP*	75
$\alpha(t)$	Number of messages that arrive at the system during t	18
$\alpha_n(s)$	LST for attained service time and the number of arrivals	18
β	Expectation of A_n for n	92
β	Rate of the underlying Markov process for *SPP*	75
η_n	State of the nth Markov point	38
γ	Throughput	17
κ	Invariant probability vector of K	93
$\lambda_\clubsuit$	Arrival rate ($\clubsuit = 1, 2, i, m$)	15
μ	Service rate	22
ω_k	Probability distribution of the number of messages just after the vacation termination point	38
π	Stationary vector of $C + D$	8
π_k	Probability distribution of the number of messages just after the Markov point	16
ρ	Offered load	17
ρ'	Carried load	17
σ	Reciprocal of the mean length of the interval between consecutive imbedded points	38
$\bar{\theta}_K$	Mean busy period for $M/G/1/K$	19
$\theta^*_K(s)$	LST for the busy period of $M/G/1/K$	20
$\varphi_n(s)$	LST for attained vacation time and the number of arrivals	40
ξ	Server state at arbitrary instant	39
ζ	State of the underlying Markov process for *SPP*	75
a_k	Probability that there are k arriving messages in a service time	16
b	Mean service time	16
$b^{(2)}$	Second moment of the service time	22
e	Column vector of ones	8
f_k	Probability that there are k arriving messages in a vacation time	38
g	Invariant probability vector of G	93
$h(x)$	pdf of the service time	76
p_{jk}	Transition probability	16
v	Vacation rate	44
$v(x)$	pdf of the vacation time	76
$\boldsymbol{x}_n^{\mathrm{s}}$	Joint probability vector of the number of messages at the service termination point	99
$\boldsymbol{x}_n^{\mathrm{v}}$	Joint probability vector of the number of messages at the vacation termination point	99
$\boldsymbol{x}_k$	Joint probability vector at departure epoch	91
$\boldsymbol{y}_k$	Joint probability vector at arbitrary instant	95
$\boldsymbol{y}_k^+$	Joint probability vector that a message arrives when the server is idle, and finds k waiting messages	96

Appendix B

M/G/1/K System with and without Vacations

The Derivation of $\alpha_n(s)$

Let X be the service time observed by an arbitrary message, then the probability density function of X is given by

$$Prob[x < X < x + dx] = \frac{xdS(x)}{b}. \tag{B.1}$$

Let $\widehat{S}$ denote an elapsed service time and $\widetilde{S}$ a remaining service time. $\widehat{S}$ and $\widetilde{S}$ satisfy

$$X = \widehat{S} + \widetilde{S}. \tag{B.2}$$

Given that $X = x$, the distribution of $\widetilde{S}$ becomes

$$E[e^{-s\widetilde{S}}|X = x] = \frac{1 - e^{-sx}}{sx}. \tag{B.3}$$

Using (B.1), (B.2), and(B.3), we obtain the generating function of $\{\alpha_n(s)\}$ as

$$\begin{aligned}
\sum_{n=0}^{\infty} \alpha_n(s) z^n &= E[e^{-s\widetilde{S}} \cdot e^{\lambda \widehat{S} z} e^{-\lambda \widehat{S}}] \\
&= \int_0^{\infty} E[e^{-s\widetilde{S} - \lambda(1-z)\widehat{S}}|X = x] \frac{xdS(x)}{b} \\
&= \int_0^{\infty} E[e^{-(s-\lambda+\lambda z)\widetilde{S}}|X = x] e^{-\lambda(1-z)x} \frac{xdS(x)}{b} \\
&= \frac{1}{(s - \lambda + \lambda z)b} \int_0^{\infty} [e^{-\lambda(1-z)x} - e^{-sx}] dS(x) \\
&= \frac{S^*(\lambda - \lambda z) - S^*(s)}{(s - \lambda + \lambda z)b} \\
&= \frac{1}{(\lambda - s)b} \left[S^*(s) \sum_{n=0}^{\infty} \left(\frac{\lambda}{\lambda - s} \right)^n z^n \right. \\
&\qquad \left. - \sum_{n=0}^{\infty} z^n \sum_{m=0}^{n} a_m \left(\frac{\lambda}{\lambda - s} \right)^{n-m} \right].
\end{aligned} \tag{B.4}$$

Above equation leads to

$$\alpha_n(s) = \frac{1}{\rho}\left[S^*(s)\left(\frac{\lambda}{\lambda - s}\right)^{n+1} - \sum_{m=0}^{n} a_m \left(\frac{\lambda}{\lambda - s}\right)^{n-m+1}\right], \quad n = 0, 1, 2, \cdots. \tag{B.5}$$

Appendix C

Waiting Time Distribution under FCFS

C.1 M/G/1/K

When k $(1 \leq k \leq K-1)$ messages are in the system, the waiting time of a new message is the remaining service time $\tilde{S}$ plus $k-1$ service times. Therefore, we obtain the $W^*(s)$ for FCFS as

$$W^*(s) = \frac{1}{1-P_B}\left[P_0 + \sum_{k=1}^{K-1} \Pi_k^*(s)\{S^*(s)\}^{k-1}\right]. \tag{C.1}$$

C.2 M/G/1/K with multiple vacations

Each message arrives either in a vacation period or in a busy period. Thus, we consider each case separately.

1. The server is on vacation.
   When k $(0 \leq k \leq K-1)$ messages are in the system, the waiting time of a new message is the remaining vacation time $\tilde{V}$ plus k service times.

2. The server is busy.
   When k $(1 \leq k \leq K-1)$ messages are in the system, the waiting time of a new message is the remaining service time $\tilde{S}$ plus $k-1$ service times.

Therefore, we obtain the $W^*(s)$ for FCFS as

$$W^*(s) = \frac{1}{1-P_B}\left[\sum_{k=0}^{K-1} \Omega_k^*(s)\{S^*(s)\}^k + \sum_{k=1}^{K-1} \Pi_k^*(s)\{S^*(s)\}^{k-1}\right]. \tag{C.2}$$

Appendix D

Waiting Time Distribution for Non-Vacation Case

In this appendix, we show the results of LSTs of the waiting time distribution for the $M/G/1/K$ with push-out scheme under non-vacation [Lee84, Rubi88].

In the non-vacation case, we choose a set of imbedded Markov points at those epochs when a service is completed. Then, we define the following limiting probability distributions.

$$\pi_k \equiv \lim_{n\to\infty} Prob[L_n = k], \qquad k = 0, 1, 2, \cdots, K-1, \tag{D.1}$$

where L_n is the number of messages in the system just after the service completion point. The set $\{\pi_k; 0 \le k \le K-1\}$ satisfies the following equations

$$\pi_k = \pi_0 a_k + \sum_{j=1}^{k+1} \pi_j a_{k-j+1}, \qquad 0 \le k \le K-2, \tag{D.2}$$

$$\pi_{K-1} = \pi_0 \left(1 - \sum_{j=0}^{K-2} a_k\right) + \sum_{j=1}^{K-1} \pi_j \left(1 - \sum_{k=0}^{K-j-1} a_k\right), \tag{D.3}$$

$$\sum_{j=0}^{K-1} \pi_k = 1. \tag{D.4}$$

From above equations, we can determine the values of $\{\pi_k\}$.

Let $\Pi_k(x)$ denote the joint probability distribution that the queue length is k and the remaining service time is less than x at an arbitrary time. Let $\Pi_k^*(s)$ denote the LST of $\Pi_k(x)$. Using $\{\pi_k\}(0 \le k \le K-1)$, we obtain LSTs as

$$\Pi_k^*(s) = \frac{1}{\pi_0 + \rho}\left[S^*(s)\left\{\pi_0\left(\frac{\lambda}{\lambda - s}\right)^k + \sum_{j=1}^{k} \pi_j \left(\frac{\lambda}{\lambda - s}\right)^{k-j+1}\right\} - \sum_{j=0}^{k-1} \pi_j \left(\frac{\lambda}{\lambda - s}\right)^{k-j}\right], \qquad 1 \le k \le K-1, \tag{D.5}$$

$$\Pi_K^*(s) = -\frac{1}{(\pi_0 + \rho)s}\left[S^*(s)\left\{\pi_0\left(\frac{\lambda}{\lambda - s}\right)^{K-1} + \sum_{j=1}^{K-1} \pi_j \left(\frac{\lambda}{\lambda - s}\right)^{K-1}\right\} - \sum_{j=0}^{K-1} \pi_j \left(\frac{\lambda}{\lambda - s}\right)^{K-j-1}\right]. \tag{D.6}$$

where $\rho = \lambda/\mu$. Using (4.10), we obtain the LST of the waiting time of a served message under FCFS as

$$W^*(s) = \pi_0 + (\pi_0 + \rho)\left[\sum_{j=1}^{K-1}\left\{\sum_{k=0}^{K-j-1}\Pi^*_{j:k}(s)\cdot W^*_{j-1:k}(s) + \sum_{k=K-j}^{K-2}\Pi^*_{j:k}(s)\cdot W^*_{K-k-2:k}(s)\right\} + \sum_{k=0}^{K-2}\Pi^*_{K:k}(s)\cdot W^*_{K-k-2:k}(s)\right], \quad \text{(D.7)}$$

where

$$\Pi^*_{j:k}(s) = \int_0^\infty \frac{(\lambda x)^k}{k!}e^{-(s+\lambda)x}d\Pi_j(x), \qquad 1 \le j \le K. \quad \text{(D.8)}$$

Using (4.14) and (4.17), we also obtain the LST of the waiting time under LCFS as

$$W^*(s) = \pi_0 + \rho\sum_{j=0}^{K-2}\widetilde{S}^*_j(s)\cdot\widetilde{W}^*_j(s). \quad \text{(D.9)}$$

Appendix E

SPP/G/1 System with Multiple Vacations and E-limited Service Discipline

E.1 Derivation of Equation (5.55)

First, we calculate $\widehat{A}^M(z)$. $\widehat{A}(z)$ are expressed as

$$\widehat{A}(z) = \begin{pmatrix} S^*(p(z)) - \hat{p}(z)\hat{q}(z)S^*(q(z)) & \hat{q}(z)(S^*(q(z)) - S^*(p(z)) \\ \hat{p}(z)(S^*(p(z)) - S^*(q(z))) & S^*(q(z)) - \hat{p}(z)\hat{q}(z)S^*(p(z)) \end{pmatrix}. \tag{E.1}$$

This matrix has the following eigenvalues and eigenvectors;

$$\begin{array}{lccc} \text{eigenvalue :} & (1-\hat{p}(z)\hat{q}(z))S^*(p(z)) & , & (1-\hat{p}(z)\hat{q}(z))S^*(q(z)), \\ \text{eigenvector :} & (1, \hat{p}(z)) & , & (\hat{q}(z), 1). \end{array}$$

In the following equations, $p(z)$, $q(z)$, $\hat{p}(z)$ and $\hat{q}(z)$ are described as p, q, $\hat{p}$ and $\hat{q}$, respectively. Then $\widehat{A}^M(z)$ is expressed as

$$\begin{aligned} \widehat{A}^M(z) &= (1-\hat{p}\hat{q})^{M-1} \begin{pmatrix} 1 & \hat{q} \\ \hat{p} & 1 \end{pmatrix} \begin{pmatrix} \{S^*(p)\}^M & 0 \\ 0 & \{S^*(q)\}^M \end{pmatrix} \begin{pmatrix} 1 & -\hat{q} \\ -\hat{p} & 1 \end{pmatrix} \\ &= (1-\hat{p}\hat{q})^{M-1} \begin{pmatrix} \{S^*(p)\}^M - \hat{p}\hat{q}\{S^*(q)\}^M & \hat{q}(\{S^*(q)\}^M - \{S^*(p)\}^M) \\ \hat{p}(\{S^*(p)\}^M - \{S^*(q)\}^M) & \{S^*(q)\}^M - \hat{p}\hat{q}\{S^*(p)\}^M \end{pmatrix}. \end{aligned} \tag{E.2}$$

Thus, $\widehat{B}(z)\widehat{A}^M(z)$ is given by

$$\widehat{B}(z)\widehat{A}^M(z) = (1-\hat{p}\hat{q})^M \begin{pmatrix} f_M(p) - \hat{p}\hat{q}f_M(q) & \hat{q}(f_M(q) - f_M(p)) \\ \hat{p}(f_M(p) - f_M(q)) & f_M(q) - \hat{p}\hat{q}f_M(p) \end{pmatrix}, \tag{E.3}$$

where $f_m(z) = V^*(z)\{S^*(z)\}^m$. Therefore, we obtain

$$\begin{aligned} &z^M(1-\hat{p}\hat{q})^{M+1}I - \widehat{B}(z)\widehat{A}^M(z) = \\ &(1-\hat{p}\hat{q})^M \begin{pmatrix} z^M(1-\hat{p}\hat{q}) - f_M(p) + \hat{p}\hat{q}f_M(q) & \hat{q}(f_M(p) - f_M(q)) \\ \hat{p}(f_M(q) - f_M(p)) & z^M(1-\hat{p}\hat{q}) - f_M(q) + \hat{p}\hat{q}f_M(p) \end{pmatrix}. \end{aligned} \tag{E.4}$$

Thus, the inverse of $z^M(1-\hat{p}\hat{q})^{M+1}I - \widehat{B}(z)\widehat{A}^M(z)$ is given by

$$\left[z^M(1-\hat{p}\hat{q})^{M+1}I - \widehat{B}(z)\widehat{A}^M(z)\right]^{-1} = \frac{1}{(1-\hat{p}\hat{q})^{M+2}(z^M - f_M(p))(z^M - f_M(q))} \cdot \begin{pmatrix} z^M(1-\hat{p}\hat{q}) - f_M(q) + \hat{p}\hat{q}f_M(p) & \hat{q}(f_M(q) - f_M(p)) \\ \hat{p}(f_M(p) - f_M(q)) & z^M(1-\hat{p}\hat{q}) - f_M(p) + \hat{p}\hat{q}f_M(q) \end{pmatrix}. \quad \text{(E.5)}$$

E.2 Proof of the Existence of the Roots of $a_p(z)$ and $a_q(z)$

First, we show that $a_p(z) = 0$, i.e.,

$$z^M - V^*(p(z))\{S^*(p(z))\}^M = 0, \quad \text{(E.6)}$$

has M roots in a unit circle $|z| \leq 1$ [Taka91].

We define $f(z)$ and $g(z)$ by

$$f(z) = z^M, \quad \text{(E.7)}$$

$$g(z) = -V^*(p(z))\{S^*(p(z))\}^M. \quad \text{(E.8)}$$

Substituting $z = z_0 + \Delta z$ into (E.8), where $|z_0| = 1$ and $|\Delta z| \ll 1$, we have

$$g(z_0 + \Delta z) = g(z_0) + \Delta z \left(\frac{dg(z)}{dz}\right)_{z=z_0} + o(\Delta z), \quad \text{(E.9)}$$

and

$$|g(z_0 + \Delta z)| \leq |g(z_0)| + \left|\Delta z \left(\frac{dg(z)}{dz}\right)_{z=z_0}\right| + o(\Delta z). \quad \text{(E.10)}$$

After some calculations, we obtain

$$|g(z_0)| \leq 1, \quad \text{(E.11)}$$

$$\left|\left(\frac{dg(z)}{dz}\right)_{z=z_0}\right| \leq \Lambda(E[V] + ME[S]). \quad \text{(E.12)}$$

Thus, (E.10) becomes

$$|g(z_0 + \Delta z)| \leq 1 + \Lambda(E[V] + ME[S])\Delta z + o(\Delta z). \quad \text{(E.13)}$$

Therefore, on $|z| = 1 + \varepsilon$ for a real and small ε, we have

$$|g(z)| \leq 1 + \Lambda(E[V] + ME[S])\varepsilon + o(\varepsilon). \quad \text{(E.14)}$$

Similarly, from (E.7), on $|z| = 1 + \varepsilon$, we have

$$|f(z)| = (1+\varepsilon)^M = 1 + M\varepsilon + o(\varepsilon). \quad \text{(E.15)}$$

Hence, if

$$\rho + \Lambda E[V]/M < 1, \quad \text{(E.16)}$$

then, $|f(z)| > |g(z)|$ on $|z| = 1+\varepsilon$. By Rouché's theorem, $f(z)$ and $f(z)+g(z)$ have the same number of zeros inside $|z| = 1+\varepsilon$. Clearly, $f(z)$ has M zeros inside $|z| = 1+\varepsilon$. Therefore, (E.6) has M roots inside $|z| = 1+\varepsilon$. One of them is

$$\omega_0 = 1. \tag{E.17}$$

The other $M-1$ roots are given from the Lagrange's theorem by

$$\omega_m = \sum_{n=1}^{\infty} \frac{e^{2\pi mni}}{n!} \left(\frac{d^{n-1}}{dz^{n-1}} \left[V^*(p(z)) \{S^*(p(z))\}^M \right]^{\frac{n}{M}} \right)_{z=0}, \qquad (1 \le m \le M-1), \tag{E.18}$$

where $i^2 = -1$.

Next, we show that $a_q(z) = 0$, i.e.

$$z^M - V^*(q(z)) \{S^*(q(z))\}^M = 0 \tag{E.19}$$

has M roots. For $|z| \le 1$,

$$|q(z)| \ge q(1) = \alpha + \beta > 0. \tag{E.20}$$

Hence, we have

$$\begin{aligned} |V^*(q(z)) \{S^*(q(z))\}^M| &= \left| \int_0^\infty e^{-q(z)t} dV(t) * H^{(M)}(t) \right| \\ &\le \int_0^\infty |e^{-q(z)t}| dV(t) * H^{(M)}(t) \\ &\le \int_0^\infty e^{-(\alpha+\beta)t} dV(t) * H^{(M)}(t) \\ &< 1, \end{aligned}$$

where $*$ denotes the convolution operator and $H^{(M)}(t)$ denotes the M-fold convolution of $H(t)$ with itself. Following the argument given above for $a_p(z)$, (E.19) has M roots in a unit circle $|z| \le 1$. Thus, from Lagrange's theorem, M roots of (E.19) are given by

$$\theta_m = \sum_{n=1}^{\infty} \frac{e^{2\pi mni}}{n!} \left(\frac{d^{n-1}}{dz^{n-1}} \left[V^*(q(z)) \{S^*(q(z))\}^M \right]^{\frac{n}{M}} \right)_{z=0}, \qquad (0 \le m \le M-1). \tag{E.21}$$

E.3 Calculation of $\psi_k^{(l)}$

In this appendix, we determine the $2M$ unknown values $\psi_k^{(l)}$ $(0 \le k \le M-1,\ l = 1,2)$ [Ozaw90]. First, we calculate the $2M-1$ unknown values $\omega_m (1 \le m \le M-1)$ and $\theta_m (0 \le m \le M-1)$. These $2M-1$ distinct roots are calculated by solving the following equations;

$$\omega_m : z - e^{\frac{2\pi(M-m)i}{M}} [V^*(p(z))]^{\frac{1}{M}} S^*(p(z)) = 0, \qquad (1 \le m \le M-1), \tag{E.22}$$

$$\theta_m : z - e^{\frac{2\pi(M-m)i}{M}} [V^*(q(z))]^{\frac{1}{M}} S^*(q(z)) = 0, \qquad (0 \le m \le M-1). \tag{E.23}$$

For each m, from Rouché's theorem, both equations have exactly one root in a unit circle $|z| \le 1$. Hence, we can calculate it using numerical methods (for example, Newton's method and the binary method).

Next, from (5.63), (5.65) and (5.66), we have following equations

$$\frac{E[V]}{M(1-\rho)-\Lambda E[V]}\sum_{k=0}^{M-1}(M-k)(\psi_k^{(1)}+\psi_k^{(2)})=1, \tag{E.24}$$

$$\sum_{k=0}^{M-1}\left\{\Omega_k^{(1)}(\omega_i)\psi_k^{(1)}+\Omega_k^{(2)}(\omega_i)\psi_k^{(2)}\right\} = 0, \qquad (1\le i\le M-1), \tag{E.25}$$

$$\sum_{k=0}^{M-1}\left\{\Theta_k^{(1)}(\theta_i)\psi_k^{(1)}+\Theta_k^{(2)}(\theta_i)\psi_k^{(2)}\right\} = 0, \qquad (0\le i\le M-1), \tag{E.26}$$

where

$$\Omega_k^{(1)}(z) = z^k\left[z^{M-k}-\{S^*(p(z))\}^{M-k}\right], \tag{E.27}$$

$$\Omega_k^{(2)}(z) = -z^k\left[z^{M-k}-\{S^*(p(z))\}^{M-k}\right]\hat{q}(z), \tag{E.28}$$

$$\Theta_k^{(1)}(z) = -z^k\left[z^{M-k}-\{S^*(q(z))\}^{M-k}\right]\hat{p}(z), \tag{E.29}$$

$$\Theta_k^{(2)}(z) = z^k\left[z^{M-k}-\{S^*(q(z))\}^{M-k}\right]. \tag{E.30}$$

We note that ω_i and ω_{M-i} or θ_i and θ_{M-i} are conjugate complex numbers. Thus, we obtain following equations

$$\sum_{k=0}^{M-1}\left\{Re[\Omega_k^{(1)}(\omega_i)]\psi_k^{(1)}+Re[\Omega_k^{(2)}(\omega_i)]\psi_k^{(2)}\right\} = 0, \qquad (1\le i\le \lfloor M/2\rfloor), \tag{E.31}$$

$$\sum_{k=0}^{M-1}\left\{Im[\Omega_k^{(1)}(\omega_i)]\psi_k^{(1)}+Im[\Omega_k^{(2)}(\omega_i)]\psi_k^{(2)}\right\} = 0, \qquad (1\le i\le \lfloor (M-1)/2\rfloor), \tag{E.32}$$

$$\sum_{k=0}^{M-1}\left\{Re[\Theta_k^{(1)}(\theta_i)]\psi_k^{(1)}+Re[\Theta_k^{(2)}(\theta_i)]\psi_k^{(2)}\right\} = 0, \qquad (1\le i\le \lfloor M/2\rfloor), \tag{E.33}$$

$$\sum_{k=0}^{M-1}\left\{Im[\Theta_k^{(1)}(\theta_i)]\psi_k^{(1)}+Im[\Theta_k^{(2)}(\theta_i)]\psi_k^{(2)}\right\} = 0, \qquad (1\le i\le \lfloor (M-1)/2\rfloor), \tag{E.34}$$

where $\lfloor x\rfloor$ means the maximum integer that does not exceed x. (E.24), (E.31), (E.32), (E.33) and (E.34) are $2M$ linearly independent equations in terms of $\psi_k^{(l)}$. Hence, we can determine $\psi_k^{(l)}$'s from those equations.

Appendix F

MAP/G/1 Queues under N-policy

Derivation of Equation (6.32)

In this appendix, we derive (6.32) using (6.27), (6.28) and (6.29). From (6.27), we obtain

$$\boldsymbol{X}^{\mathrm{s}}(z)\left[zI - A(z)\right] = \boldsymbol{X}^{\mathrm{v}}(z)A(z) - \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)\right] A(z). \tag{F.1}$$

Using (6.28) and (F.1), we have

$$\boldsymbol{X}^{\mathrm{s}}(z)\left[zI - A(z)\right] = \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(z)\right]\left[V(z) - I\right] A(z).$$

Substituting $z = 1$ and multiplying both sides of (6.28) by $\boldsymbol{e}$, we obtain

$$\boldsymbol{X}^{\mathrm{v}}(1)\boldsymbol{e} = \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e}. \tag{F.2}$$

It then follows from (6.29) and (F.2) that

$$\boldsymbol{X}^{\mathrm{s}}(1)\boldsymbol{e} = 1 - \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e}. \tag{F.3}$$

Next, setting $z = 1$ and adding $\boldsymbol{X}^{\mathrm{s}}(1)\boldsymbol{e}\boldsymbol{\pi}$ to both sides of (F.1), we have

$$\begin{aligned} \boldsymbol{X}^{\mathrm{s}}(1) &= \boldsymbol{X}^{\mathrm{s}}(1)\boldsymbol{e}\boldsymbol{\pi} + \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]\left[V - I\right]A\left(I - A + \boldsymbol{e}\boldsymbol{\pi}\right)^{-1} \\ &= \left(1 - \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e}\right)\boldsymbol{\pi} \\ &\quad + \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]\left[V - I\right]A\left(I - A + \boldsymbol{e}\boldsymbol{\pi}\right)^{-1}. \end{aligned} \tag{F.4}$$

Multiplying both sides of (F.4) by $A'(1)\boldsymbol{e}$, we obtain

$$\begin{aligned} \boldsymbol{X}^{\mathrm{s}}(1)A'(1)\boldsymbol{e} &= \rho\left[1 - \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e}\right] \\ &\quad + \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]\left[V - I\right](A - \boldsymbol{e}\boldsymbol{\pi})(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e} \\ &= \rho\left[1 - \left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e}\right] \\ &\quad + \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]\left[V - I\right]A(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e}, \end{aligned} \tag{F.5}$$

where we use the equality

$$A'(1)\boldsymbol{e} = \rho\boldsymbol{e} + (I - A)(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e}.$$

On the other hand, differentiating (F.1) and setting $z = 1$ yield

$$\begin{aligned} \boldsymbol{X}^{\mathrm{s}}(1)[I - A'(1)]\boldsymbol{e} &= \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]AV'(1)\boldsymbol{e} \\ &= \lambda E[V]\left(\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right)\boldsymbol{e} \\ &\quad + \left[\boldsymbol{x}_0^{\mathrm{s}} + \boldsymbol{X}_{N-1}^{\mathrm{v}}(1)\right]\left(I - V\right)A(\boldsymbol{e}\boldsymbol{\pi} - C - D)^{-1}D\boldsymbol{e}, \end{aligned} \tag{F.6}$$

where we use the equality

$$V'(1)e = \lambda E[V]e + (I - V)(e\pi - C - D)^{-1}De.$$

Thus, it follows from from (F.5) and (F.6) that

$$\boldsymbol{X}^{s}(1)e = \lambda E[V]\left(x_0^{s} + \boldsymbol{X}_{N-1}^{v}(1)\right)e + \rho\left[1 - \left(x_0^{s} + \boldsymbol{X}_{N-1}^{v}(1)\right)e\right]. \tag{F.7}$$

Finally, using (F.3) and (F.7), we obtain

$$\left(x_0^{s} + \boldsymbol{X}_{N-1}^{v}(1)\right)e = \frac{1-\rho}{1-\rho+\lambda E[V]}. \tag{F.8}$$

References

[Asmu93] Asmussen, S. and Koole, G., "Marked Point Processes as Limits of Markovian Arrival Streams," *Journal of Applied Probability*, Vol. 30, pp.365–372, 1993.

[Armb87] Armbruster, H. and G. Arndt, "Broadband Communication and Its Realization with Broadband ISDN," *IEEE Communications Magazine*, Vol. 25, No. 11, pp.8–19, 1987.

[Blon91] Blondia, C., "Finite Capacity Vacation Models with Non-renewal input," *Journal of Applied Probability*, Vol. 28, pp.174–197, 1991.

[Coop81] Cooper, R. B., *Introduction to Queuing Theory, Second Edition*, Elsevier, Amsterdam, 1981.

[Dosh86] Doshi, B. T., "Queueing Systems with Vacations – A Survey," *Queueing Systems*, Vol. 1, pp.29–66, 1986.

[Dosh90] Doshi, B. T., "Single Server Queues with Vacations," *Stochastic Analysis of Computer and Communication System*, Elsevier Science Publishers B. V., North–Holland, pp.217–265, 1990.

[Fisc92] Fischer, W. and Meier-Hellstern, K., "The Markov-modulated Poisson Process (*MMPP*) Cookbook," *Performance Evaluation*, Vol. 18, 149–171, 1992.

[Fuhr85] Fuhrmann, S. W. and Cooper, R. B., "Stochastic Decompositions in the $M/G/1$ Queue with Generalized Vacations," *Operations Research*, Vol. 33, pp.1117–1129, 1985.

[Grun91] Grünenfelder, R., Cosmas, J. P., Manthrope, S. and Odinma-Okafor, A., "Characterization of Video Codecs as Autoregressive Moving Average Processes and Related Queueing Performance," *IEEE Journal on Selected Areas in Communications*, Vol. 9, pp.284–293, 1991.

[Heff86] Heffes, H. and Lucantoni, D. M., "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance," *IEEE Journal on Selected Areas in Communications*, Vol. 4, pp.856–868, 1986.

[Heym82] Heyman, D. P. and Sobel, M. J., *Stochastic Models in Operations Research Volume I: Stochastic Processes and Operating Characteristics*, McGraw-Hill, New York, 1982.

[Hofi86] Hofri, M., "Queueing Systems with a Procrastinating Server," *Performance '86 and ACM SIGMETRICS 1986, Performance Evaluation Review*, Vol. 14, No. 1, pp.245–253, 1986.

[Kasa89] Kasahara, S., Takahashi, Y. and Hasegawa, T., "Analysis of Waiting Time of $M/G/1/K$ System under Random Scheduling and LCFS," Graduation Thesis of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University, 1989.

[Kasa93a] Kasahara, S., Takagi, H., Takahashi, Y. and Hasegawa, T., "$M/G/1/K$ System with Push-out Scheme under vacation Policy," *Telecommunication Systems Conference: Modeling and Analysis*, Nashville, Tennessee, February 28 – March 3, 1993.
To be appeared in Journal of Applied Mathematics and Stochastic Analysis.

[Kasa93b] Kasahara, S., Takine, T., Takahashi, Y. and Hasegawa, T., "Analysis of an $SPP/G/1$ System with Multiple Vacations and E-limited Service Discipline," *Queueing Systems*, Vol. 14, pp.349–367, 1993.

[Kasa95a] Kasahara, S., Takahashi, Y. and Hasegawa, T., "Analysis of Waiting Time of $M/G/1/K$ System with Vacations under Random Scheduling and LCFS," *Performance Evaluation*, Vol. 21, pp.239–259, 1995.

[Kasa95b] Kasahara, S., Takine, T., Takahashi, Y. and Hasegawa, T., "$MAP/G/1$ Queues under N-policy with and without Vacations," *To be appeared in Journal of the Operations Research Society of Japan.*

[Kell89] Kella, O., "The Threshold Policy in the $M/G/1$ Queue with Server Vacations," *Naval Research Logistics*, Vol. 36, pp.111–123, 1989.

[Klei75] Kleinrock, L., *Queueing Systems Volume1: Theory*, John Wiley & Sons, New York, 1975.

[Krön90] Kröner, H., "Comparative Performance Study of Space Priority Mechanisms for ATM Networks," *IEEE INFOCOM'90*, pp.1136–1143, 1990.

[Lee84] Lee, T. T., "$M/G/1/N$ Queue with Vacation Time and Exhaustive Service Discipline," *Operations Research*, Vol. 32, No. 4, pp.774–784, 1984.

[Lee89a] Lee, T. T., "$M/G/1/N$ Queue with Vacation Time and Limited Service Discipline," *Performance Evaluation*, Vol. 9, No. 3, pp.181–190, 1989.

[Lee89b] Lee, H. and Srinivasan, M. M., "Control Policies for the $M^X/G/1$ Queueing System," *Management Science*, Vol. 35, No. 6, pp.708–721, 1989.

[Luca90] Lucantoni, D. M., Meier-Hellstern, K. and Neuts, M. F., "A Single Server Queue with Server Vacations and a Class of Non-renewal Arrival Processes," *Advances in Applied Probability*, Vol. 22, pp.676–705, 1990.

[Luca91] Lucantoni, D M., "New Results on the Single Server Queue with a Batch Markovian Arrival Process," *Stochastic Models.*, Vol 7, No 1, pp.1–46, 1991.

[Luca93] Lucantoni, D. M., "The $BMAP/G/1$ Queue: A Tutorial," *Models and Techniques for Performance Evaluation of Computer and Communication Systems*, Donatiello, L. and Nelson, R. (eds.), Springer Verlag, pp.330–358, 1993.

[Neut79] Neuts, M. F., "A Versatile Markovian Point Process," *Journal of Applied Probability*, Vol. 16, pp.764–779, 1979.

[Neut81] Neuts, M. F., *Matrix-Geometric Solutions in Stochastic Models*, The Johns Hopkins University Press, Baltimore and London, 1981.

[Neut89] Neuts, M. F., *Structured Stochastic Matrices of $M/G/1$ Type and their Applications*, Marcel Dekker, INC, New York, 1989.

[Ozaw90] Ozawa, T., "Alternating Service Queues with Mixed Exhaustive and K-limited Services," *Performance Evaluation*, Vol. 11, pp.165–175, 1990.

[Part94] Partridge, C., *Gigabit Networking*, Addison-Wesley, Massachusetts, 1994.

[Rama80] Ramaswami, V., "The $N/G/1$ Queue and its Detailed Analysis," *Advances in Applied Probability*, Vol. 12, pp.222–261, 1980.

[Rama88] Ramaswami, V., "Stable Recursion for the Steady State Vector in Markov Chains of $M/G/1$ Type," *Stochastic Models.*, Vol. 4, pp.183–188, 1988.

[Rubi88] Rubin, I., and Ouaily, M., "Performance of Finite Capacity Communication and Queueing Systems under Various Service and Buffer Preemption Policies," *IEEE INFOCOM'88*, pp.505–514, 1988.

[Seng89] Sengupta, B., "Markov Processes Whose Steady State Distribution is Matrix-exponential with an Application to the $GI/PH/1$ Queue," *Advances in Applied Probability*, Vol. 21, pp.159–180, 1989.

[Seng90] Sengupta, B., "The Semi-Markovian Queue: Theory and Applications," *Stochastic Models.*, Vol. 6, No. 3, pp.383–413, 1990.

[Sumi88] Sumita, S. and Ozawa, T., "Achievability of Performance Objectives in ATM Switching Nodes," *Performance of Distributed and Parallel Systems*, North-Holland, Amsterdam, pp.45–56, 1988.

[Taká63] Takács, L., "Delay Distributions for One Line with Poisson Input, General Holding Times and Various Orders of Service," *The Bell System Technical Journal*, Vol. 42, pp.487–503, March 1963.

[Taka85] Takagi, H., "Analysis of a Finite-Capacity $M/G/1$ Queue with a Resume Level," *Performance Evaluation*, Vol. 5, pp.197–203, 1985.

[Taka89] Takagi, H., "Queueing Analysis of Vacation Models Part 5 : $M/G/1/K$," Working Paper, IBM Tokyo Research Laboratory, Tokyo, Japan, 1989.

[Taka91] Takagi, H., *Queueing Analysis: A Foundation of Performance Evaluation Volume 1: Vacation and Priority Systems, Part 1*, North-Holland, Amsterdam, 1991.

[Taka93] Takagi, H., *Queueing Analysis: A Foundation of Performance Evaluation Volume 2: Finite Systems*, North-Holland, Amsterdam, 1993.

[Taki93a] Takine, T. and Hasegawa, T., "A Batch $SPP/G/1$ Queue with Multiple Vacations and Exhaustive Service Discipline," *Telecommunication Systems*, Vol. 1, pp.195–215, 1993.

[Taki93b] Takine, T., Matsumoto, Y., Suda, T. and Hasegawa, T., "Mean Waiting Times in Nonpreemptive Priority Queues with Markovian Arrival and I.I.D. Service Processes," *Proceeding of Performance '93*, Rome, Italy, pp. 129-148, 1993.

[Taki94a] Takine, T. and Takahashi, Y., "On the Relationship between Queue Lengths at a Random Instant and at a Departure in the Stationary Queue with *BMAP* Arrivals," in preparation.

[Taki94b] Takine, T., "Introduction to $M/G/1$ Paradigm and $MAP/G/1$ Queue," Working Paper, 1994.

[Turn86] Turner, J. S., "New Directions in Communications (or Which Way to the Information Age?)," *IEEE Communications Magazine*, Vol. 24, No. 10, pp.8–15, 1986.

[Wolf82] Wolff, R. W., "Poisson Arrivals See Time Averages," *Operations Research*, Vol. 30., pp.223–231, 1982.

[Wolf89] Wolff, R. W., *Stochastic Modeling and the Theory of Queues*, Prentice Hall, Inc., New Jersey, 1989.